

Rayan BAH

Mathieu GAUDE

Bastien BONGIORNO

Guillaume BARREAULT

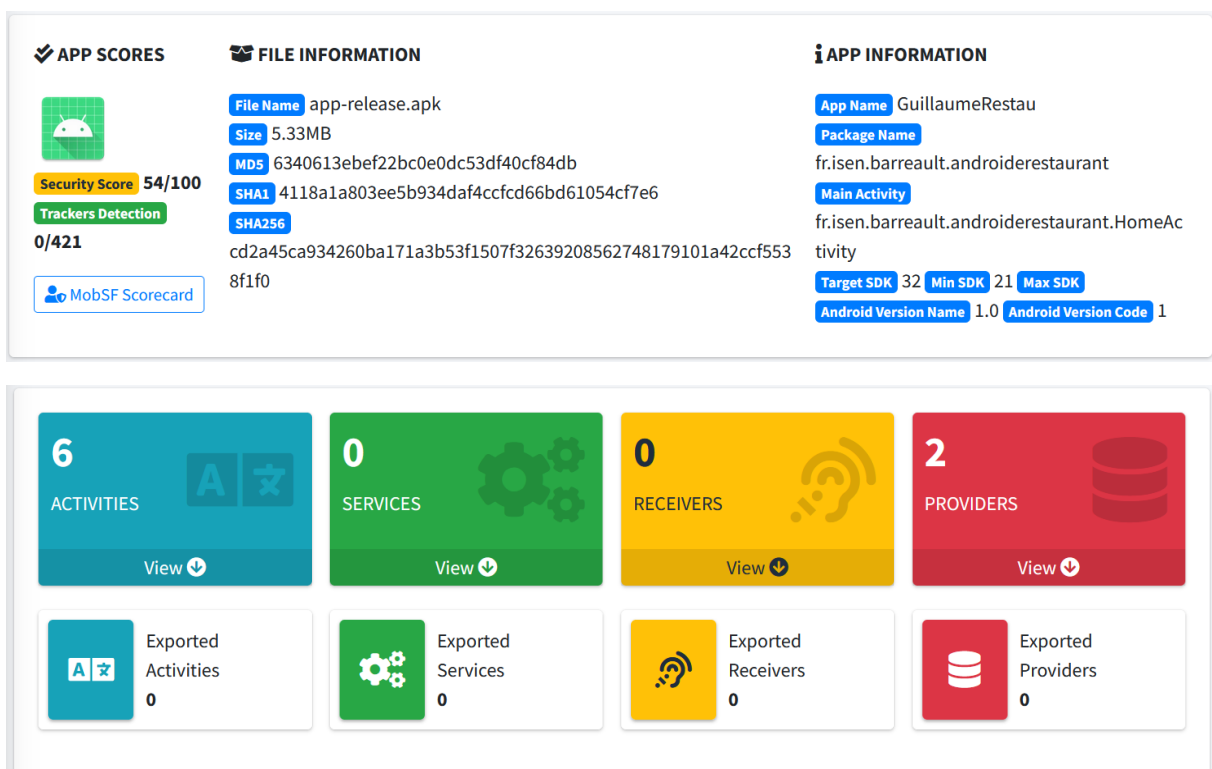
Thomas MEUNIER

Projet de Sécurité Mobile

Pour le projet de sécurité mobile, nous avons utilisé MobSf pour analyser et améliorer la sécurité de notre application de restauration en fonction des données et du score donné par le site.

Premier Scan :

Pour notre tout premier test, le score est de 54/100.



Lors du premier test, nous avons une visualisation de la signature en V1 et V2 et des différents hashes.

SIGNER CERTIFICATE

```
APK is signed
v1 signature: True
v2 signature: True
v3 signature: False
Found 1 unique certificates
Subject: CN=ISEN
Signature Algorithm: rsassa_pkcs1v15
Valid From: 2022-02-14 11:18:30+00:00
Valid To: 2047-02-08 11:18:30+00:00
Issuer: CN=ISEN
Serial Number: 0x7dbe99ef
Hash Algorithm: sha256
md5: 0d30b1308445e6ad14a85807f2a02e1b
sha1: 244a9569e8bfb1df5e17735385ebc57058da6316
sha256: f69dd3c8320974bfbdb85864b2e74a0222f94bed4e734727ae1d4fa983ec4a43a
sha512: 4f1c8f9aee5b9b0ed8f35b7a512ea8d4da549c7edb79b47239fda3c713870ce8aebce5aa678049fb3d3cb132e079d90f8d9d9cc0916e86c48
PublicKey Algorithm: rsa
Bit Size: 2048
Fingerprint: 37286eb47414f5a1ce142f2bbaa225df8a348cfc4dafa716020b3e437361b3d4
```





L'application est signée avec le schéma de signature v1, ce qui la rend vulnérable à l'attaque Janus sur les versions Android de 5.0 à 8.0. Les applications fonctionnant sur Android 5.0 - 7.0 sont vulnérables qu'elles soient signées avec v1, v2 ou v3. Quant à la version Android 8.0, elle est vulnérable uniquement lorsqu'elle est signée avec v1.

Cette vulnérabilité a été patché sur les versions supérieures d'Android (9 et +).

Voici les autres données analysées pour l'application :

APPLICATION PERMISSIONS

Search:

PERMISSION 	STATUS 	INFO 	DESCRIPTION 
android.permission.ACCESS_NETWORK_STATE	normal	view network status	Allows an application to view the status of all networks.
android.permission.INTERNET	normal	full Internet access	Allows an application to create network sockets.

Showing 1 to 2 of 2 entries

Previous **1** Next

Q MANIFEST ANALYSIS			
Search: <input type="text"/>			
NO ↑↓	ISSUE ↑↓	SEVERITY ↑↓	DESCRIPTION ↑↓
1	Clear text traffic is Enabled For App [android:usesCleartextTraffic=true]	high	The app intends to use cleartext network traffic, such as cleartext HTTP, FTP stacks, DownloadManager, and MediaPlayer. The default value for apps that target API level 27 or lower is "true". Apps that target API level 28 or higher default to "false". The key reason for avoiding cleartext traffic is the lack of confidentiality, authenticity, and protections against tampering; a network attacker can eavesdrop on transmitted data and also modify it without being detected.
2	Application Data can be Backed up [android:allowBackup=true]	warning	This flag allows anyone to backup your application data via adb. It allows users who have enabled USB debugging to copy application data off of the device.

Showing 1 to 2 of 2 entries

Previous 1 Next

Dans notre code, *CleartextTraffic* a pour valeur *"true"* ce qui signifie que l'application utilise le trafic réseau en texte clair, tel que HTTP en texte clair, les piles FTP, DownloadManager et MediaPlayer.

En effet, la valeur par défaut pour les applications qui ciblent le niveau d'API 27 ou inférieur est *"true"*. Les applications qui ciblent le niveau d'API 28 ou supérieur sont définies par défaut sur *"false"*. La principale raison d'éviter le trafic en clair est le manque de confidentialité, d'authenticité et de protection contre la falsification ; un attaquant du réseau peut espionner les données transmises et aussi les modifier sans être détecté (sniffing et spoofing).

Si on change la valeur *"CleartextTraffic"* pour *false*, le serveur distant utilisé dans l'application enverrait des requêtes HTTP, cela nous restreindrait ; nous serions dans l'obligation d'imposer le trafic HTTPS qui ne serait pas accepté par le serveur.

Ce problème nous a donc empêché de fixer la vulnérabilité.


Enfin, la vulnérabilité de Data Backup. Cela permet de sauvegarder et restaurer l'état de l'application par synchronisation data dans le cloud, on peut donc les retrouver dans le cache.

De ce fait, l'intégralité des données stockées dans le dossier d'une application (/data/data/com.example.app) sont donc visibles ce qui constitue un réel danger pour la sécurité des informations, sans oublier le fait qu'il n'y ait pas besoin d'être root

Avec la commande *debuggable*, on peut décider si l'application peut être déboguée ou non, même lorsqu'elle est exécutée sur un appareil en mode utilisateur. La valeur est à *"true"* si elle peut l'être, et à *false* sinon. La valeur par défaut est *"false"*.

Il est suggéré par la documentation officielle de Microsoft de rendre l'application *non debuggable*, c'est une mesure de sécurité pour éviter la reverse-engineering ou le vol de données d'application.

Deuxième Scan :



Security Score 72/100

Trackers Detection 0/421

[MobSF Scorecard](#)

FILE INFORMATION

File Name app-release.apk

Size 3.72MB

MD5 f4c7411fd31acaf5d175ab8527f65e0b

SHA1 db5f64eb6e95001734afc8a2d84fdc027c3ef81

SHA256 9a2b467185d3b8c8f7c00638ffaa690a84692a88cc4c91eeafb6457039e11f56

Après avoir supprimé la vulnérabilité du AllowBackup ainsi qu’avoir augmenté la valeur du SDK dans le *build.gradle*, le score a bien augmenté.

```
defaultConfig {
    applicationId "fr.isen.barreault.androiderestaurant"
    minSdk 30
    targetSdk 32
    versionCode 1
    versionName "1.0"

    testInstrumentationRunner "androidx.test.runner.AndroidJUnitRunner"
}
```

Cependant, nous avons une donnée nous parlant d’un générateur de nombre aléatoire. Après vérification, nous nous sommes rendu compte que cela venait de l’obfuscation du code.

high Clear text traffic is Enabled For App	MANIFEST
medium Files may contain hardcoded sensitive information like usernames, passwords, keys etc.	CODE
medium The App uses an insecure Random Number Generator.	CODE
info The App logs information. Sensitive information should never be logged.	CODE
info App can write to App Directory. Sensitive Information should be encrypted.	CODE
secure This App uses SSL certificate pinning to detect or prevent MITM attacks in secure communication channel.	CODE
secure This application has no privacy trackers	TRACKERS

```
buildTypes {
    release {
        minifyEnabled true
        shrinkResources true
        proguardFiles getDefaultProguardFile('proguard-and
    }
}
```

L'obfuscation est très importante pour empêcher la compréhension du code par l'attaquant. Nous avons activé l'obfuscation avec Proguard. Ceci rentre en compte dans le MSTG-RESILIENCE-9 qui correspond à du **R**. Cela permet aux fonctions et aux fichiers d'avoir des noms aléatoires incompréhensibles pour un être humain mais toujours exécutables pour un ordinateur. Pour l'activer, nous avons ajouté ces lignes dans le code qui permettent d'activer Proguard et d'obfusquer par défaut. Nous avons aussi ajouté une commande dans le Proguard. *Overload aggressively*. Plusieurs champs et méthodes peuvent alors obtenir les mêmes noms, tant que leurs arguments et leurs types de retour sont différents, comme l'exige le **Bytecode Java**. Cette option peut rendre le code traité encore plus petit (et moins compréhensible). Cela améliore grandement la sécurité de notre application, cependant, cette option ajoute un générateur aléatoire qui n'est pas très sécurisé.

S'il n'y a pas d'obfuscation, le score diminue mais MobSf n'affiche plus le générateur de nombres aléatoires.

APP SCORES

Security Score **56/100**

Trackers Detection **0/421**

[MobSF Scorecard](#)

FILE INFORMATION

File Name app-release.apk

Size 11.11MB

MD5 5a5f53b62f0519c5b027bb2b54ca7aa4

SHA1 b9439c2cef0361e6847733e963f69ef1063e1c47

SHA256 e9f365f202f43247e3d7ff2cb9b44d6ebe39ae3ce651baf77c7d0624ab8d6118

APP INFORMATION

App Name GuillaumeRestau

Package Name fr.isen.barreault.androiderestaurant

Main Activity fr.isen.barreault.androiderestaurant.HomeActivity

Target SDK 32 Min SDK 30 Max SDK 30

Android Version Name 1.0 Android Version Code 1

High 1

Medium 1

Info 2

Secure 1

Hotspot 0

high Clear text traffic is Enabled For App [MANIFEST](#)

medium Files may contain hardcoded sensitive information like usernames, passwords, keys etc. [CODE](#)

info The App logs information. Sensitive information should never be logged. [CODE](#)

info App can write to App Directory. Sensitive Information should be encrypted. [CODE](#)

secure This application has no privacy trackers [TRACKERS](#)

Tentatives non abouties :

- Tentative de lancer un scan dynamique sur MobSf :

Cela génère une erreur nous signifiant que l'émulateur n'est pas activé alors que l'émulateur est bien lancé et nous avons testé avec deux autres émulateurs. Nous pensons que cela vient d'une erreur en fonction de l'émulateur utilisé.

- Tentative d'implémentation Anti-Frida

En recompilant l'application on obtient des erreurs. Nous avons donc enlevé toutes les modifications faites liées à notre *anti-frida*.

- **Burp :**

Nous avons utilisé ce logiciel pour analyser ce que l'application envoyée à l'API et obtenir un meilleur contrôle de la gestion de mot de passe (donc au moment de l'inscription ou de l'authentification). Nous avons réussi à voir que le mot de passe était bien haché.

Nous avons aussi remarqué que lorsque que l'application est lancée et que l'utilisateur est authentifié, le hash est envoyé mais lorsque on essaye de s'identifier une nouvelle fois directement avec un autre compte, le hash du mot de passe du 2nd compte est le même que le mot de passe du 1er compte.

Pour pallier ce problème l'application doit être réinitialiser l'application, donc la redémarrer.

- **Shared Preferences :**

Nous avons tenté de chiffrer les préférences partagées du contenu présent dans le panier (par exemple le nombre d'item) mais nous avons eu des erreurs lors de compilation.

Rapport pour le serveur :

Du côté du serveur, il y a plusieurs aspects à développer et implémenter.

Il y a d'abord le *ClearText Traffic*. En le passant à *False*, la communication entre le serveur (c'est-à-dire l'API) et l'application passe en HTTPS. Or, l'API ne gère que l'HTTP donc en passant en HTTPS les ressources de celle-ci ne pourraient plus être disponibles sur l'application. Pour pallier cela, il faudrait configurer le serveur pour l'adapter et qu'il autorise les communications en HTTPS.

De plus, pour le stockage du mot de passe, le serveur récupère le mot de passe hashé en SHA-256. Une implémentation possible serait de hashé avec de l'argon le hash du SHA-256 auxquelles serait ajouté un sel aléatoire. Puis ce nouveau hash serait chiffré avec de l'AES-256. Enfin dans la Database serait stocké le sel aléatoire et le hash chiffré; et dans un key.config la clé AES

Améliorations Futures :

Pour les améliorations possibles dans le futur, nous ciblerons 3 principaux problèmes : le Clear Text Traffic, le générateur de nombre aléatoire et la détection de Frida.

Pour le Clear Text Traffic, cela signifie que l'application communique avec le serveur en clair c'est-à-dire en http. Pour supprimer cette vulnérabilité et passer en https, nous devrions rajouter la ligne `useClearTextTraffic=True`. L'API ne gère que l'http donc en passant en https nous n'aurions plus pu communiquer avec.

Le serveur devrait donc s'adapter et autoriser l'https.

Comme dit précédemment, l'obfuscation a généré une vulnérabilité, un générateur de nombre aléatoire non sécurisé. En effectuant nos recherches, nous avons conclu que ce problème devait se

trouver dans une librairie utilisée par Proguard. Nous envisageons la possibilité de pouvoir supprimer cette vulnérabilité en changeant les propriétés de ProGuard.

Pour terminer dans les implémentations futures, nous n'avons pas intégré un moyen de détecter l'utilisation de Frida pour prévenir les injections de scripts. En effet, cette détection est une solution permettrait de scanner la mémoire des librairies pour checker la présence des fichiers spécifiques de Frida.

Le problème avec la détection de Frida est dans l'implémentation en elle-même de la solution.

Main report in the Application

CWE: CWE-532: Insertion of Sensitive Information into Log File
OWASP MASVS: MSTG-STORAGE-3

CWE: CWE-330: Use of Insufficiently Random Values
OWASP Top 10: M5: Insufficient Cryptography
OWASP MASVS: MSTG-CRYPTO-6

CWE: CWE-276: Incorrect Default Permissions
OWASP MASVS: MSTG-STORAGE-14

CWE: CWE-312: Cleartext Storage of Sensitive Information
OWASP Top 10: M9: Reverse Engineering
OWASP MASVS: MSTG-STORAGE-14

✓: Valid ✗: Invalid /: Non defined in the Application

V1 : Exigences Concernant l'Architecture, le Design et le Modèle de Menaces

MSTG-ARCH-1 ✓

MSTG-ARCH-2 ✗

MSTG-ARCH-3 ✗

MSTG-ARCH-4 ✓

MSTG-ARCH-5 ✗

MSTG-ARCH-6 ✓

MSTG-ARCH-7 ✗

MSTG-ARCH-8 ✓

MSTG-ARCH-9 ✗

MSTG-ARCH-10 ✗

MSTG-ARCH-11 ✓

MSTG-ARCH-12 ✓

V2 : Exigences Concernant le Stockage des Données et le Respect de la Vie Privée

MSTG-STORAGE-1 ✓

MSTG-STORAGE-2 /

MSTG-STORAGE-3 ✓

MSTG-STORAGE-4 ✓

MSTG-STORAGE-5 ✓

MSTG-STORAGE-6 ✗

MSTG-STORAGE-7 ✗

MSTG-STORAGE-8 ✓

MSTG-STORAGE-9 ✗

MSTG-STORAGE-10 ✗

MSTG-STORAGE-11 /

MSTG-STORAGE-12 /

MSTG-STORAGE-13 /

MSTG-STORAGE-14 ✓

MSTG-STORAGE-15 /

V3 : Exigences Concernant la Cryptographie

MSTG-CRYPTO-1 ✓

MSTG-CRYPTO-2 ✓

MSTG-CRYPTO-3 ✓

MSTG-CRYPTO-4 ✓

MSTG-CRYPTO-5 ✓

MSTG-CRYPTO-6 ✗

V4 : Exigences Concernant l'Authentification et la Gestion des Sessions

MSTG-AUTH-1 ✗

MSTG-AUTH-2 ✗

MSTG-AUTH-3 ✗

MSTG-AUTH-4 ✗

MSTG-AUTH-5 ✓

MSTG-AUTH-6/

MSTG-AUTH-7/

MSTG-AUTH-8 ✗

MSTG-AUTH-9 ✗

MSTG-AUTH-10 ✗

MSTG-AUTH-11 ✗

MSTG-AUTH-12 ✗

V5 : Exigences Concernant la Communication Réseau

MSTG-NETWORK-1 ✗

MSTG-NETWORK-2 ✗

MSTG-NETWORK-3 ✗

MSTG-NETWORK-4 ✗

MSTG-NETWORK-5 ✗

MSTG-NETWORK-6 ✗

Aucune main mise sur le serveur et le réseau

V6 : Exigences Concernant les Interactions avec la Plateforme

Non implémenté dans l'application

V7 : Exigences Concernant la Qualité du Code et les Paramètres de Génération

MSTG-CODE-1 ☒

MSTG-CODE-2 ☒

MSTG-CODE-3 ☐

MSTG-CODE-4 ☒

MSTG-CODE-5 ☒

MSTG-CODE-6 ☐

MSTG-CODE-7 ☐

MSTG-CODE-8 ☐

MSTG-CODE-9 ☐

V8 : Exigences Concernant la Résilience

MSTG-RESILIENCE-1 ☒

MSTG-RESILIENCE-2 ☒

MSTG-RESILIENCE-3 ☐

MSTG-RESILIENCE-4 ☐

MSTG-RESILIENCE-5 ☐

MSTG-RESILIENCE-6 ☐

MSTG-RESILIENCE-7 ☐

MSTG-RESILIENCE-8 ☐

MSTG-RESILIENCE-9 ☒

MSTG-RESILIENCE-10 ☐

MSTG-RESILIENCE-11 ☒

MSTG-RESILIENCE-12 /

MSTG-RESILIENCE-13 ☐