

ANALYSE DU GESTIONNAIRES DE MOTS DE PASSE

✓ REALISEATION :

- EL BEKHTI AYOUB
- HAIMOUD YASSINE
- ISMAIL ARAME

✓ ENCADRÉ PAR:

- Mr.KOULALI MOHAMMED AMINE

❖ **Presentation de projet et leur but:**

Dans ce monde de plus en plus connecté, où les données personnelles sont devenues une monnaie d'échange virtuelle, la sécurité des informations sensibles est cruciale. Les cybermenaces évoluent constamment, des attaques sophistiquées de phishing aux vulnérabilités zero-day, menaçant la confidentialité et l'intégrité des données.

Les applications de gestionnaires de mots de passe se positionnent comme une première ligne de défense essentielle dans la protection des informations personnelles. En consolidant l'accès derrière un seul point d'entrée sécurisé, ces applications réduisent le risque d'exposition due à des pratiques de gestion de mots de passe inadéquates, telles que l'utilisation de mots de passe faibles ou la réutilisation de mots de passe sur plusieurs plateformes.

Dans ce projet on souhaite de faire une analyse dynamique et statique de certaines gestionnaires de mots de passe et tester leur security , mais avant faire ca c'est quoi un gestionnaire de mots passe ?

Un gestionnaire de mots de passe est une application conçue pour aider les utilisateurs à gérer et à stocker de manière sécurisée leurs informations d'identification en ligne, telles que les noms d'utilisateur et les mots de passe. Ces outils sont particulièrement utiles à mesure que les utilisateurs accumulent un grand nombre de comptes sur différents services en ligne.

➤ POUR QUOI ON UTILISE LES GESTIONNAIRES DE MOTS DE PASSE ?

Les gestionnaires de mots de passe sont utilisés pour plusieurs raisons importantes, visant à améliorer la sécurité et la gestion des informations d'identification en ligne. Voici quelques-unes des principales raisons pour lesquelles les gens utilisent des gestionnaires de mots de passe :

- **Complexité des Mots de Passe** : Les recommandations en matière de sécurité suggèrent l'utilisation de mots de passe complexes, difficiles à deviner. Cependant, la gestion de nombreux mots de passe complexes peut être difficile sans assistance. Les gestionnaires de mots de passe permettent de générer, stocker et gérer automatiquement des mots de passe forts.
- **Sécurité Renforcée** : Les gestionnaires de mots de passe utilisent souvent des techniques de cryptage avancées pour sécuriser les informations d'identification. Cela ajoute une couche de protection supplémentaire par rapport à la mémorisation manuelle des mots de passe ou à la réutilisation de mots de passe sur plusieurs comptes.
- **Facilité d'Utilisation** : Les gestionnaires de mots de passe simplifient le processus d'authentification en ligne. Les utilisateurs n'ont pas besoin de se souvenir de plusieurs mots de passe, car le gestionnaire peut remplir automatiquement les champs d'identification.
- **Gestion de Nombreux Comptes** : À mesure que les utilisateurs s'inscrivent à de plus en plus de services en ligne, la gestion des informations d'identification devient une tâche complexe.

❖ Les outils qu'on va utiliser :

▪ JADX :

est un outil open-source utilisé pour décompiler des fichiers DEX (Dalvik Executable) issus d'applications Android (.apk) en fichiers Java source. Les fichiers DEX sont le résultat de la compilation des fichiers sources Java vers le format exécutable pour la machine virtuelle Dalvik, qui était utilisée par Android avant l'adoption de la machine virtuelle ART (Android Runtime).



▪ BURPSUITE :

Burp Suite est une suite d'outils de sécurité des applications web développée par PortSwigger. C'est un ensemble d'outils utilisés par les professionnels de la sécurité informatique pour effectuer des tests d'intrusion et des analyses de sécurité sur les applications web.



```
uri; window.addEventListener("blur", function() { if (window.clickbandit.mouseover) { hideButton(); ("mouseover"); setTimeout(function() { generateClickArea(++window.clickbandit.clickCount, 1000); ("mouseout"); } ); } }); var uri = new UriBuilder(uri).setHost(backendURL.getHost()).setPort(backendURL.getPort()).setScheme(backendURL.getScheme()).build();
```

- **FRIDA :**

Frida est un framework open-source permettant le dynamic instrumentation d'applications, ce qui signifie qu'il offre la possibilité de manipuler et de modifier le comportement d'applications en cours d'exécution sur divers systèmes d'exploitation. Il est souvent utilisé dans le domaine de la sécurité informatique pour effectuer des tests d'intrusion, des analyses de sécurité, et pour la recherche en sécurité des applications.

FRIDA

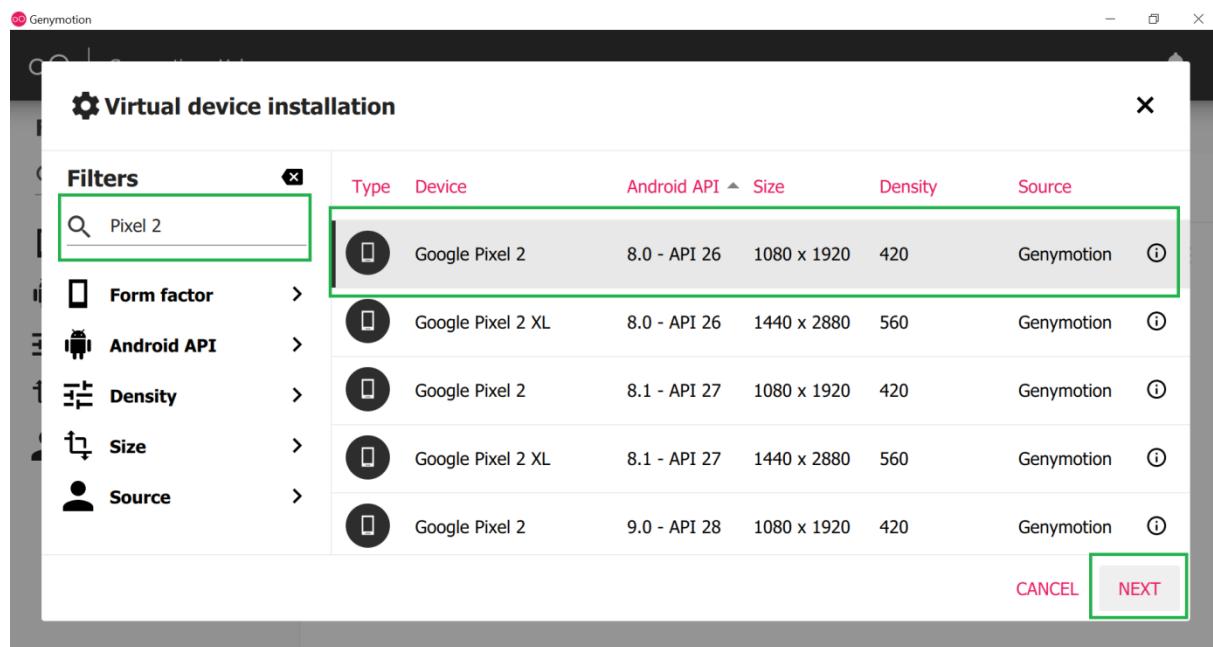
- **OBJECTION :**

est un runtime mobile en temps réel, alimentée par Frida, conçue pour vous aider à évaluer la posture de sécurité de vos applications mobiles sans avoir besoin d'un jailbreak.



- **GENYMOTION :**

Genymotion est un émulateur Android puissant et polyvalent conçu pour simplifier le processus de développement et de test d'applications Android. Il offre un environnement virtuel permettant d'exécuter des applications Android sur différents dispositifs virtuels.



- **ADB :**

ADB (Android Debug Bridge) est une interface de ligne de commande qui fait partie du kit de développement Android (Android SDK). Elle permet une communication entre un ordinateur et un appareil Android, que ce soit un smartphone, une tablette, ou un autre dispositif.



❖ **Les applications android qu'on va analyser :**

Keeper : Keeper provides useful extra features like an encrypted messaging service and has more customer support options.



NordPass : NordPass includes more cloud storage and uses a more “futureproof” encryption.



mSecure: basic password manager features like password sharing, two-factor authentication, a password generator, autofill, and a password strength report to help keep hackers at bay.



Bitwarden :is best for users looking for a platform with advanced security measures and insights and the most affordable pricing.

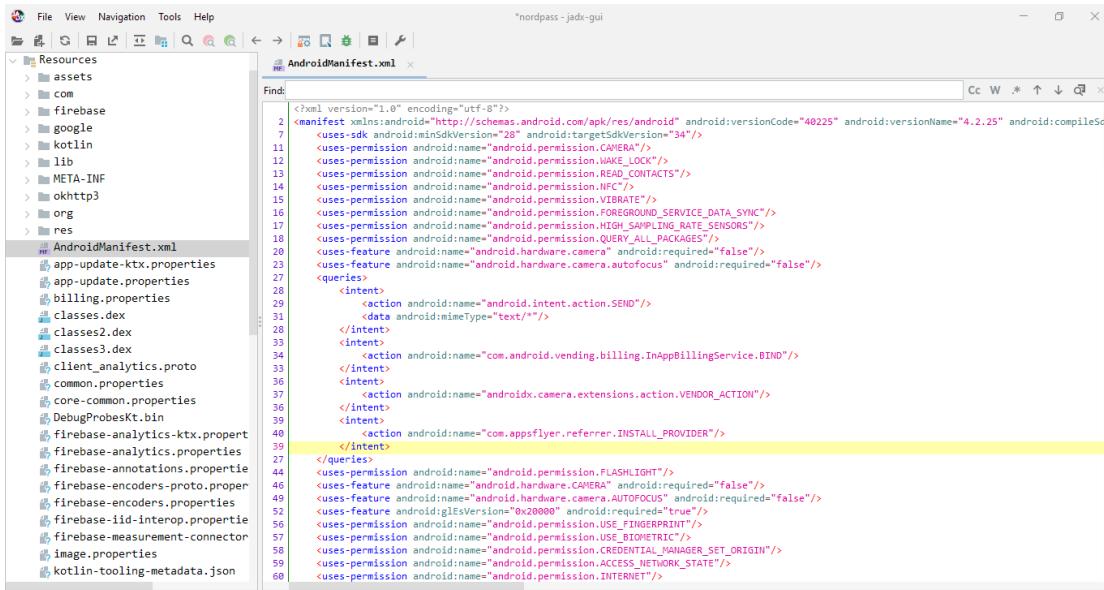


❖ Partie technique : analyse statique et dynamique des applications :

➤ NordPass :

✚ Static Analysis :

- ✓ On démarre l'exploration du fichier NordPass.apk dans Jadx, et comme d'habitude, on entame notre analyse en examinant les autorisations dans le fichier AndroidManifest.xml :



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="40225" android:versionName="4.2.25" android:compileSdkVersion="30" android:targetSdkVersion="34">
    <uses-sdk android:minSdkVersion="20" android:targetSdkVersion="34"/>
    <uses-permission android:name="android.permission.CAMERA"/>
    <uses-permission android:name="android.permission.WAKE_LOCK"/>
    <uses-permission android:name="android.permission.READ_CONTACTS"/>
    <uses-permission android:name="android.permission.NFC"/>
    <uses-permission android:name="android.permission.VIBRATE"/>
    <uses-permission android:name="android.permission.FOREGROUND_SERVICE_DATA_SYNC"/>
    <uses-permission android:name="android.permission.HIGH_SAMPLING_RATE_SENSORS"/>
    <uses-permission android:name="android.permission.QUERY_ALL_PACKAGES"/>
    <uses-feature android:name="android.hardware.camera" android:required="false"/>
    <uses-feature android:name="android.hardware.camera.autofocus" android:required="false"/>
    <queries>
        <intent>
            <action android:name="android.intent.action.SEND"/>
            <data android:mimeType="text/*"/>
        </intent>
        <intent>
            <action android:name="com.android.vending.billing.InAppBillingService.BIND"/>
        </intent>
        <intent>
            <action android:name="androidx.camera.extensions.action.VENDOR_ACTION"/>
        </intent>
        <intent>
            <action android:name="com.appsflyer.referrer.INSTALL_PROVIDER"/>
        </intent>
    </queries>
    <uses-permission android:name="android.permission.FLASHLIGHT"/>
    <uses-feature android:name="android.hardware.CAMERA" android:required="false"/>
    <uses-feature android:name="android.hardware.camera.AUTOFOCUS" android:required="false"/>
    <uses-feature android:glEsVersion="0x0009" android:required="true"/>
    <uses-permission android:name="android.permission.USE_FINGERPRINT"/>
    <uses-permission android:name="android.permission.USE_BIOMETRIC"/>
    <uses-permission android:name="android.permission.CREDENTIAL_MANAGER_SET_ORIGIN"/>
    <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
    <uses-permission android:name="android.permission.INTERNET"/>
```

- ✓ NordPass demande à chaque user de créer son propre master password qui est la clé de son coffre-fort. Grâce à lui, il autorise NordPass à réaliser sa fonction : stocker, récupérer et remplir automatiquement les mots de passe à sa place :

```
package com.nordpass.android.ui.authentication.masterpassword.create;

CreatePasswordConfirmFragment createPasswordConfirmFragment = (CreatePasswordConfirmFragment) this.f26281s;
int i12 = CreatePasswordConfirmFragment.I;
uc0.l.f(createPasswordConfirmFragment, "this$0");
CreatePasswordConfirmViewModel h11 = createPasswordConfirmFragment.h();
String str = ((go.a) createPasswordConfirmFragment.H.getValue()).f13766a;
String text = ((i3) createPasswordConfirmFragment.N()).R.getText();
h11.getClass();
uc0.l.f(str, "masterPassword");
uc0.l.f(text, "confirmMasterPassword");
h11.m(true, new go.d(h11, str, text, null));
return;
```

- ✓ Pour le chiffrement & déchiffrement, NordPass utilise l'algorithme XChaCha20 qui est un moyen de chiffrer et de déchiffrer les données. Il prend en charge deux longueurs de clés différentes ; le chiffrement 256 bits étant le plus fort. NordPass utilise XChaCha20 pour chiffrer notre coffre-fort de mots de passe :

```

43 static {
44     HashMap hashMap = new HashMap();
45     hashMap.put(Ciphers.TLS_AES_128_GCM_SHA256, "AEAD-AES128-GCM-SHA256");
46     hashMap.put(Ciphers.TLS_AES_256_GCM_SHA384, "AEAD-AES256-GCM-SHA384");
47     hashMap.put(Ciphers.TLS_CHACHA20_POLY1305_SHA256, "AEAD-CHACHA20-POLY1305-SHA256");
48     j2oTls1 = Collections.unmodifiableMap(hashMap);
49     HashMap hashMap2 = new HashMap();
50     hashMap2.put(Ciphers.TLS_AES_128_GCM_SHA256, Collections.singletonMap("TLS", Ciphers.TLS_AES_128_GCM_SHA256));
51     hashMap2.put(Ciphers.TLS_AES_256_GCM_SHA384, Collections.singletonMap("TLS", Ciphers.TLS_AES_256_GCM_SHA384));
52     hashMap2.put(Ciphers.TLS_CHACHA20_POLY1305_SHA256, Collections.singletonMap("TLS", Ciphers.TLS_CHACHA20_POLY1305_SHA256));
53     hashMap2.put("AEAD-AES128-GCM-SHA256", Collections.singletonMap("TLS", Ciphers.TLS_AES_128_GCM_SHA256));
54     hashMap2.put("AEAD-AES256-GCM-SHA384", Collections.singletonMap("TLS", Ciphers.TLS_AES_256_GCM_SHA384));
55     hashMap2.put("AEAD-CHACHA20-POLY1305-SHA256", Collections.singletonMap("TLS", Ciphers.TLS_CHACHA20_POLY1305_SHA256));
56     o2jTls13 = Collections.unmodifiableMap(hashMap2);
57 }

```

- ✓ On voit l'usage de ChaCha20-Poly1305, c'est un algorithme de chiffrement authentifié avec données supplémentaires (AEAD), qui combine le chiffrement de flux ChaCha20 avec le code d'authentification de message Poly1305. Ce dernier peut être utilisé pour authentifier un seul message à l'aide d'une clé partagée entre le client et le NordPass server. - Pour la génération et le stockage des clés , ce password manager utilise AndroidKeyStore comme un conteneur des clés qui sont générées à partir de l'algorithme AES en mode CBC:

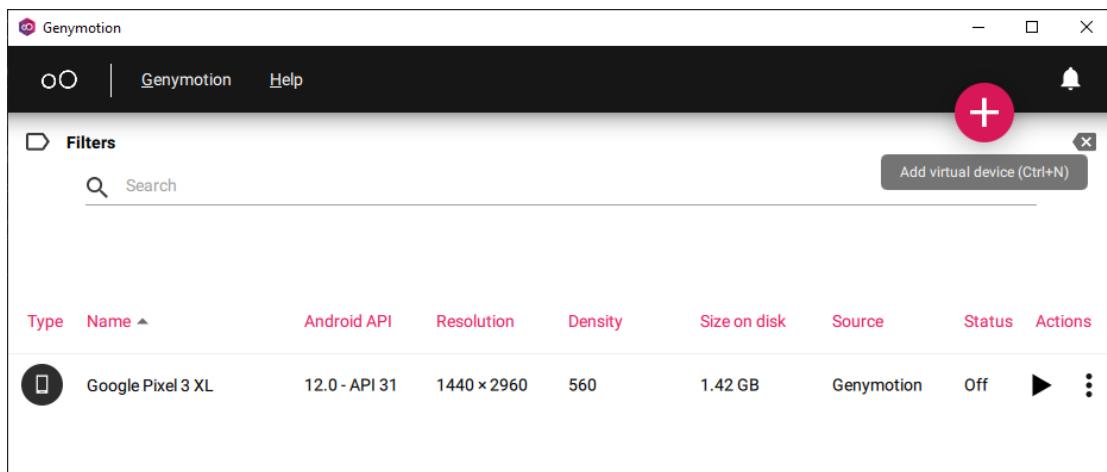
```

KeyStore keyStore = KeyStore.getInstance("AndroidKeyStore");
keyStore.load(null);
KeyGenParameterSpec.Builder b12 = y.a.b("androidxBiometric", 3);
y.a.d(b12);
y.a.e(b12);
KeyGenerator keyGenerator = KeyGenerator.getInstance("AES", "AndroidKeyStore");
y.a.c(keyGenerator, y.a.a(b12));
keyGenerator.generateKey();
Cipher cipher = Cipher.getInstance("AES/CBC/PKCS7Padding");
cipher.init(1, (SecretKey) keyStore.getKey("androidxBiometric", null));
cVar = new BiometricPrompt.c(cipher);

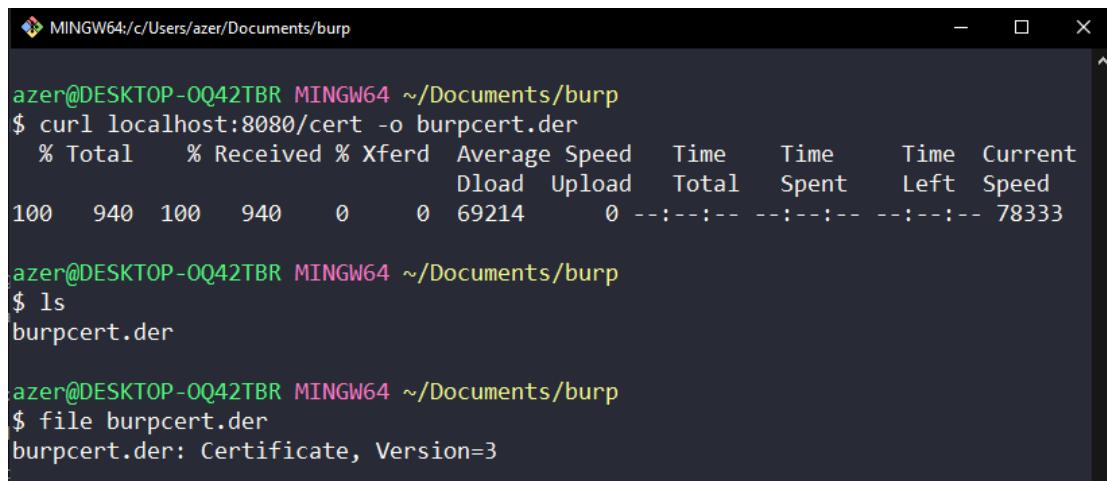
```

Dynamic Analysis :

- ✓ Configuring burp suite with genymotion :



- ✓ Tout d'abord, nous devrons générer un certificat Burp Suite pour l'interception du trafic HTTPS.

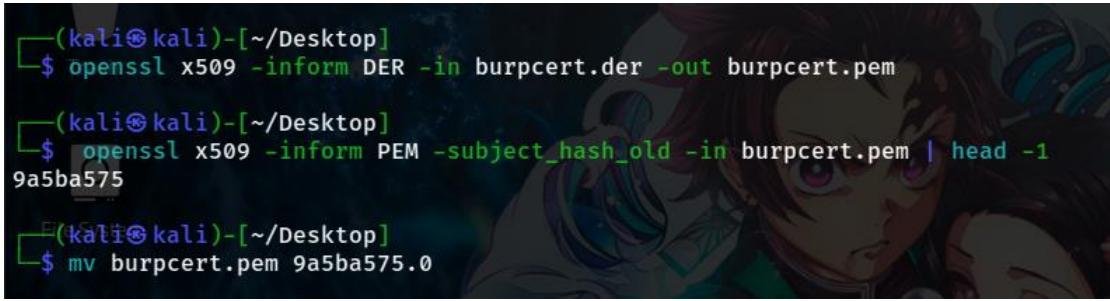


```
azer@DESKTOP-0Q42TBR MINGW64 ~/Documents/burp
$ curl localhost:8080/cert -o burpcert.der
% Total    % Received % Xferd  Average Speed   Time   Time     Current
          Dload  Upload   Total   Spent    Left  Speed
100  940  100  940    0      0  69214      0 --:--:-- --:--:-- 78333

azer@DESKTOP-0Q42TBR MINGW64 ~/Documents/burp
$ ls
burpcert.der

azer@DESKTOP-0Q42TBR MINGW64 ~/Documents/burp
$ file burpcert.der
burpcert.der: Certificate, Version=3
```

- ✓ en format der et puis on va utiliser l'outil OpenSSL pour la convertir en format PEM

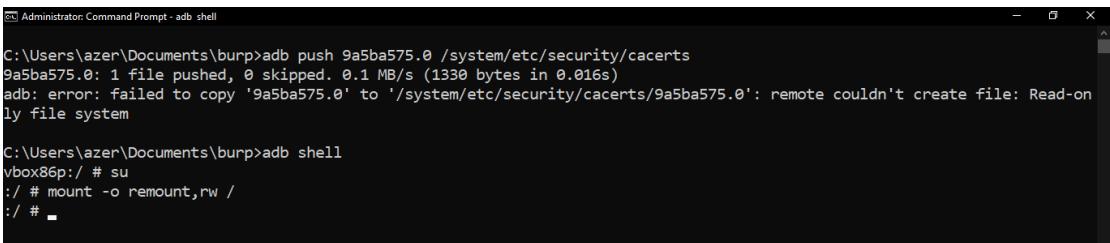


```
(kali㉿kali)-[~/Desktop]
$ openssl x509 -inform DER -in burpcert.der -out burpcert.pem

(kali㉿kali)-[~/Desktop]
$ openssl x509 -inform PEM -subject_hash_old -in burpcert.pem | head -1
9a5ba575

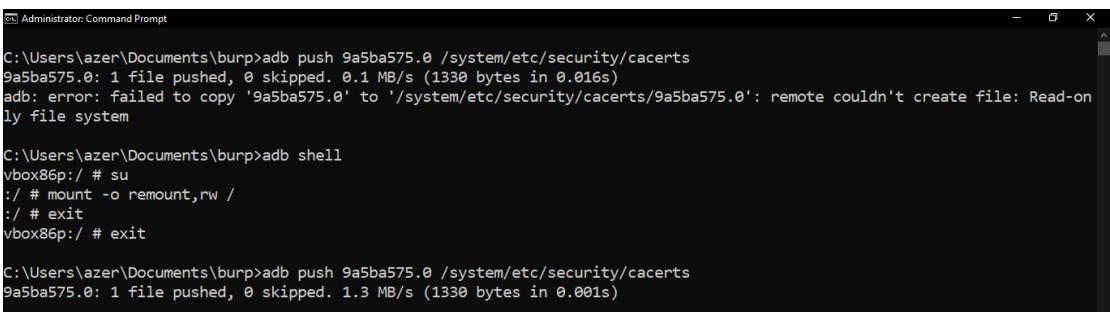
(kali㉿kali)-[~/Desktop]
$ mv burpcert.pem 9a5ba575.0
```

- ✓ On utilise la commande push d'adb afin de déplacer le fichier de la certification vers la sd card de l'émulateur.
- ✓ Mais nous n'avons pas la permission d'écrire dans le système de fichiers, d'abord nous allons le remonter en écriture et ensuite pousser le certificat à nouveau.



```
C:\Users\azer\Documents\burp>adb push 9a5ba575.0 /system/etc/security/cacerts
9a5ba575.0: 1 file pushed, 0 skipped. 0.1 MB/s (1330 bytes in 0.016s)
adb: error: failed to copy '9a5ba575.0' to '/system/etc/security/cacerts/9a5ba575.0': remote couldn't create file: Read-only file system

C:\Users\azer\Documents\burp>adb shell
vbox86p:/ # su
:/ # mount -o remount,rw /
:/ #
```

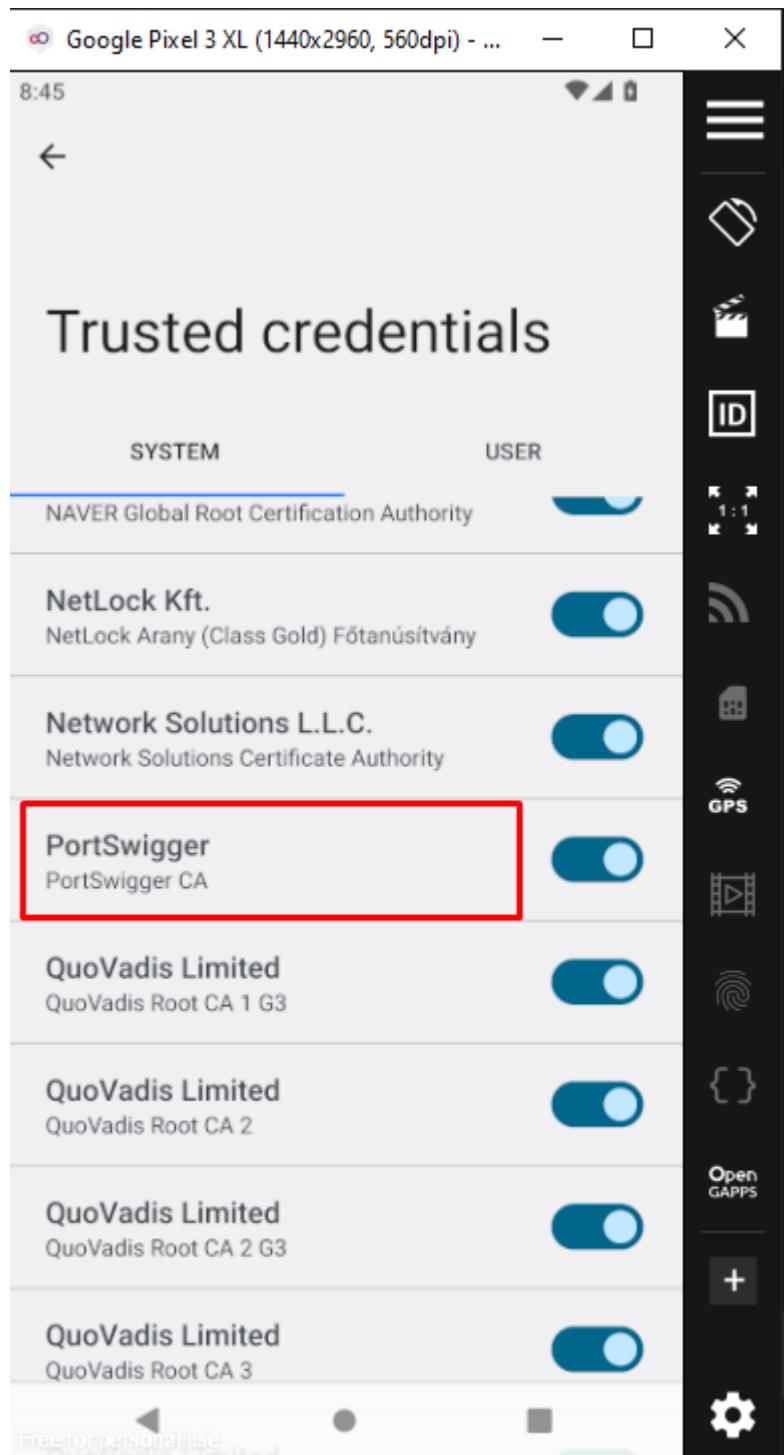


```
C:\Users\azer\Documents\burp>adb push 9a5ba575.0 /system/etc/security/cacerts
9a5ba575.0: 1 file pushed, 0 skipped. 0.1 MB/s (1330 bytes in 0.016s)
adb: error: failed to copy '9a5ba575.0' to '/system/etc/security/cacerts/9a5ba575.0': remote couldn't create file: Read-only file system

C:\Users\azer\Documents\burp>adb shell
vbox86p:/ # su
:/ # mount -o remount,rw /
:/ # exit
vbox86p:/ # exit

C:\Users\azer\Documents\burp>adb push 9a5ba575.0 /system/etc/security/cacerts
9a5ba575.0: 1 file pushed, 0 skipped. 1.3 MB/s (1330 bytes in 0.001s)
```

- ✓ Maintenant si on vérifie les certificats sur notre appareil émulé , on va retrouver celui de Burpsuite



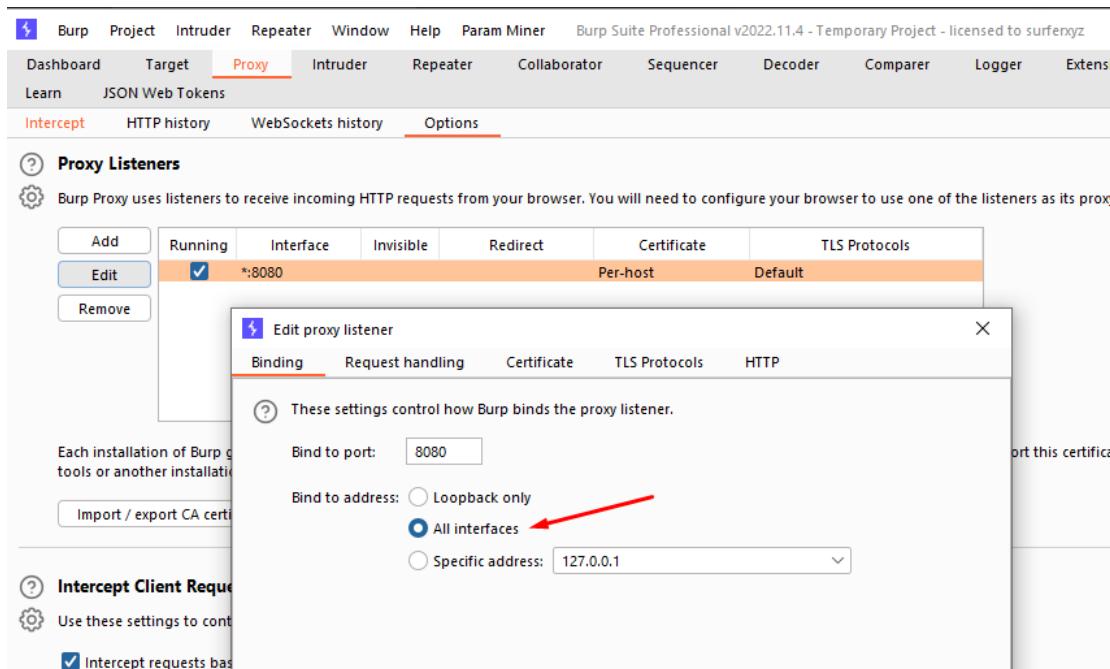
- ✓ Configurer le proxy via la ligne de commande.

```

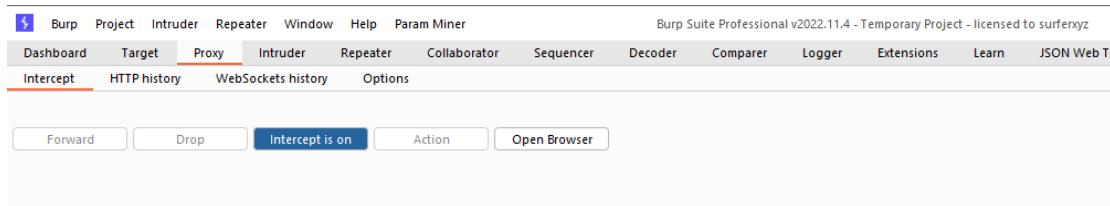
Administrator: Command Prompt
C:\Users\azer\Documents\burp>adb shell settings put global http_proxy 192.168.11.101:8080
C:\Users\azer\Documents\burp>

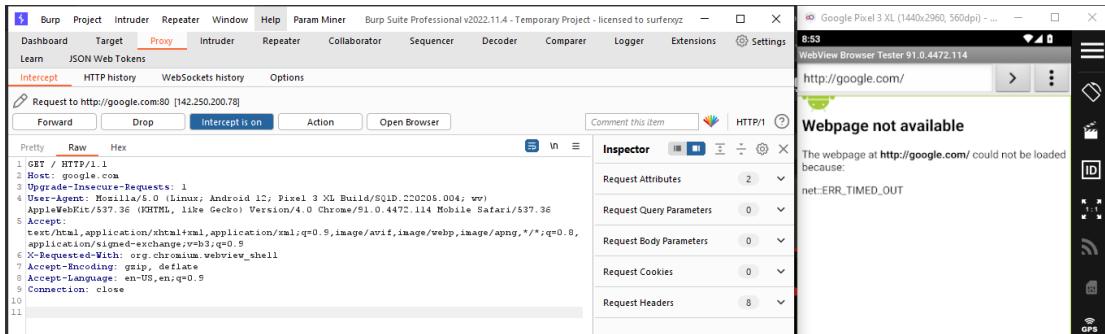
```

- ✓ Puisque Genymotion fonctionne comme une machine virtualisée, nous allons définir l'adresse **All interfaces** au lieu de **loopback** (127.0.0.1)



- ✓ Maintenant, si nous testons d'intercepter un quelconque trafic, nous remarquons que nous avons réussi.





✓ installing frida :

```
C:\Windows\system32>pip install frida
Requirement already satisfied: frida in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (16.1.7)
Requirement already satisfied: typing-extensions in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (from frida) (4.8.0)
```

```
C:\Windows\system32>pip install frida-tools
Requirement already satisfied: frida-tools in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (12.3.0)
Requirement already satisfied: colorama<1.0.0,>=0.2.7 in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (from frida-tools) (0.4.6)
Requirement already satisfied: frida<17.0.0,>=16.0.9 in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (from frida-tools) (16.1.7)
Requirement already satisfied: prompt-toolkit<4.0.0,>=2.0.0 in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (from frida-tools) (3.0.38)
Requirement already satisfied: pygments<3.0.0,>=2.0.2 in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (from frida-tools) (2.14.0)
Requirement already satisfied: typing-extensions in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (from frida<17.0.0,>=16.0.9->frida-tools) (4.8.0)
Requirement already satisfied: wccwidth in c:\users\azer\appdata\local\programs\python\python310\lib\site-packages (from prompt-toolkit<4.0.0,>=2.0.0->frida-tools) (0.2.6)
```

✓ Obtenez l'architecture du CPU pour télécharger un serveur Frida compatible

```
C:\Windows\system32>adb shell
vbox86p:/ # su
:/ # uname -a
Linux localhost 5.10.101+ genymotion+ ab74 #1 SMP PREEMPT Thu Dec 1 14:03:02 UTC 2022 x86_64
:/ #
```

✓ Execute le server frida

```
c:\Users\azer\Documents>adb push frida-server-16.1.10-android-x86_64 /data/local/tmp  
frida-server-16.1.10-android-x86_64: 1 file pushed, 0 skipped. 33.3 MB/s (108313432 bytes in 3.098s)  
  
c:\Users\azer\Documents>adb shell  
generic_x86_64_arm64:/ # su  
generic_x86_64_arm64:/ # cd /data/local/tmp  
generic_x86_64_arm64:/data/local/tmp # ls  
burp.cer frida-server-16.1.10-android-x86_64  
generic_x86_64_arm64:/data/local/tmp # chmod 777 frida-server-16.1.10-android-x86_64  
generic_x86_64_arm64:/data/local/tmp # ls -la  
total 105812  
drwxrwx--x 3 shell shell 4096 2024-01-04 10:03 .  
drwxr-x--x 4 root root 4096 2023-12-05 10:01 ..  
drwxr-xr-x 2 shell shell 4096 2024-01-04 09:34 .studio  
-rw-rw-rw- 1 shell shell 940 2024-01-03 16:13 burp.cer  
-rwxrwxrwx 1 root root 108313432 2024-01-04 09:56 frida-server-16.1.10-android-x86_64  
generic_x86_64_arm64:/data/local/tmp # ./frida-server-16.1.10-android-x86_64
```

```
C:\Windows\system32>frida-ps -U  
PID Name  
---  
2267 Google  
5839 LastPass  
5244 Settings  
6153 adb  
189 android.hardware.atrace@1.0-service  
274 android.hardware.audio.service.ranchu  
278 android.hardware.authsecret@1.0-service  
440 android.hardware.biometrics.face@1.0-service.example  
442 android.hardware.biometrics.fingerprint@2.1-service  
279 android.hardware.bluetooth@1.1-service.sim  
280 android.hardware.camera.provider@2.4-service  
281 android.hardware.camera.provider@2.6-service-google  
282 android.hardware.cas@1.2-service  
283 android.hardware.contexthub@1.1-service.mock  
284 android.hardware.drm@1.0-service  
285 android.hardware.drm@1.3-service.clearkey  
286 android.hardware.drm@1.3-service.widevine  
287 android.hardware.gatekeeper@1.0-service.software  
897 android.hardware.gnss@2.0-service.ranchu  
288 android.hardware.graphics.allocator@3.0-service  
289 android.hardware.graphics.composer@2.3-service  
290 android.hardware.health@2.1-service
```

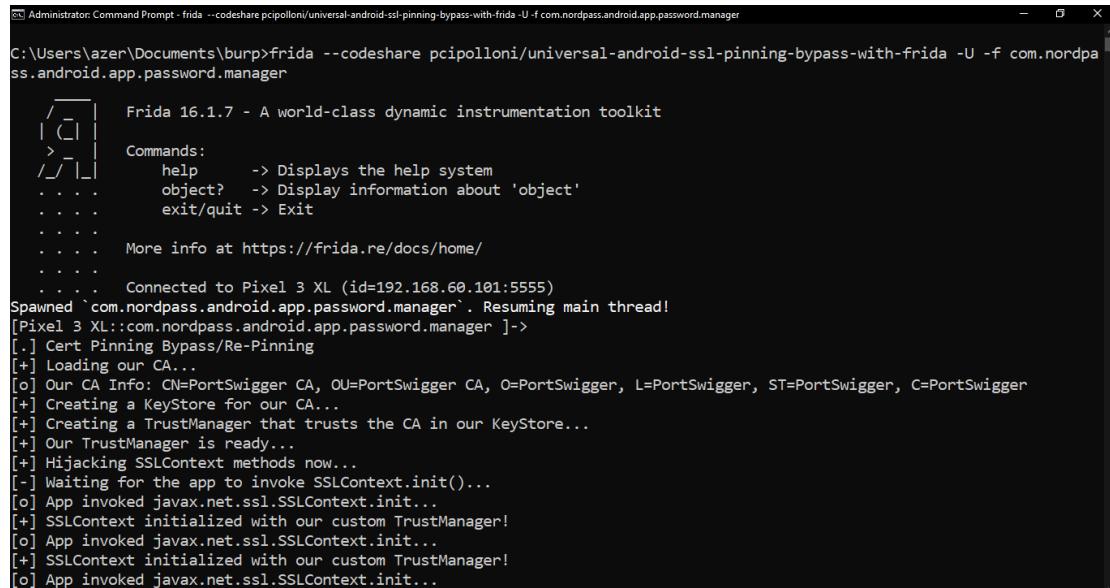
- SSL Pinning :

- ✓ On a besoin de certificat burpsuite pour le ssl pinning réussi

```
C:\Users\azer\Documents\burp>adb push burpcert.crt /data/local/tmp/cert-der.crt
burpcert.crt: 1 file pushed, 0 skipped. 0.7 MB/s (940 bytes in 0.001s)

C:\Users\azer\Documents\burp>adb shell
vbox86p:/ # su
:/ # cd /data/local/tmp
:/data/local/tmp # ls
cert-der.crt  frida-server-16.1.10-android-x86_64
:/data/local/tmp # ■
```

- ✓ Pour le SSL pinning bypass on trouve sur Frida Codeshare la commande nécessaire :
- ✓ <https://codeshare.frida.re/@pcipolloni/universal-android-ssl-pinning-bypass-with-frida/>



```
Administrator: Command Prompt - frida --codeshare pcipolloni/universal-android-ssl-pinning-bypass-with-frida -U -f com.nordpass.android.app.password.manager

C:\Users\azer\Documents\burp>frida --codeshare pcipolloni/universal-android-ssl-pinning-bypass-with-frida -U -f com.nordpass.android.app.password.manager

    _____
   |     |
   |  _` |  Frida 16.1.7 - A world-class dynamic instrumentation toolkit
   | / \ |
   | \_ \_| Commands:
   | . . . |     help      -> Displays the help system
   | . . . |     object?   -> Display information about 'object'
   | . . . |     exit/quit -> Exit
   | . . . |
   | . . . |     More info at https://frida.re/docs/home/
   | . . . |
   | . . . |     Connected to Pixel 3 XL (id=192.168.60.101:5555)
Spawner `com.nordpass.android.app.password.manager` Resuming main thread!
[Pixel 3 XL::com.nordpass.android.app.password.manager ]->
[.] Cert Pinning Bypass/Re-Pinning
[+] Loading our CA...
[+] Our CA Info: CN=PortSwigger CA, OU=PortSwigger CA, O=PortSwigger, L=PortSwigger, ST=PortSwigger, C=PortSwigger
[+] Creating a KeyStore for our CA...
[+] Creating a TrustManager that trusts the CA in our KeyStore...
[+] Our TrustManager is ready...
[+] Hijacking SSLContext methods now...
[-] Waiting for the app to invoke SSLContext.init()...
[o] App invoked javax.net.ssl.SSLContext.init...
[+] SSLContext initialized with our custom TrustManager!
[o] App invoked javax.net.ssl.SSLContext.init...
[+] SSLContext initialized with our custom TrustManager!
[o] App invoked javax.net.ssl.SSLContext.init...
```

- ✓ Et on commence par la création de notre compte sur l'application : L'application me dérige vers leur page web pour créer un compte, j'ai entrer mes informations personnelles, apres l'envoie du requette au serveur j'ai remaqué que mon password est en text clair, d'où un attacker qui entrain d'espionner va avoir l'access à mon password

Burp Suite Professional v2022.11.4 - Temporary Project - licensed to surfonyz

Request Attributes: 2

Request Query Parameters: 1

Request Body Parameters: 2

Request Cookies: 11

Request Headers: 30

Inspector

HTTP/2

Comment this item

Forward Drop Intercept is on Action Open Browser

Pretty Raw Hex

```

1 POST /signup/set-password?challenge=2%7C109dalb7c4cf43db5cda0159a371be6 HTTP/2
2 Host: nordaccount.com
3 Cookie: sessions_bag
4 MT-Auth=MT-Auth-Header|PFP8F7cbab521B1B0UF9Q5F01U2WWWzP07F9QUTFB0B_HHFFCSRRe=V1UPVodahVVE6TFRM11q3SR0R
1MT5hbdNPyecBnLwvTVJNPF1lPac-abU15W1FBPXmcHnGJH1ECC6HBRv7qs51S05vcb4gLo=xO1OMzTA+-,
dark_mode=false; locale=en; SAID=
MTcvWDJLMsk0MHMHN0GdPals=mxXNB1QzVoY1GdpFyQm51v0Z2vNmIamlyMD1GBsvwUsj0jERKBSVmTwec11sloTuS0ow4
3FIhSG6la-, request=134fde7-0e16-437b-951b-905b4c4db0bf; csrf=TlocBCWvLuJUJiURk20CyBrc0YALB5K;
nv_trs=1c04455170458_1704455774060_1_0
4 Content-Type: application/x-www-form-urlencoded
5 Cache-Control: no-store
6 Upgrade-Insecure-Requests: 1
7 Origin: https://nordaccount.com
8 Content-Type: application/x-www-form-urlencoded
9 User-Agent: Mozilla/5.0 (Linux; Android 12; Pixel 3 XL Build/SQ1D.220905.004; wv
10 AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/91.0.472.114 Mobile Safari/537.36
11 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,
application/signed-exchange;v=b3;q=0.9
12 X-Requested-With: org.chromium.webview_shell
13 Sec-Fetch-Site: same-origin
14 Sec-Fetch-Dest: navigate
15 Sec-Fetch-User: ?1
16 Referer:
https://nordaccount.com/signup/set-password?challenge=2%7C109dalb7c4cf43db5cda0159a371be6
17 Accept-Encoding: gzip, deflate
18 Accept-Language: en-US,en;q=0.9
19
20 password=sayonara12%,csrf=TlocBCWvLuJUJiURk20CyBrc0YALB5K

```

mot de pass en text clair

Could be stronger

Create password

Wrong email address? Go back

Switch to dark mode English support@nordaccount.com Terms of Service © 2024 Nord Security. All Rights Reserved.

Free for personal use

- ✓ Apres la création de mon compte sur website, j'ai passé au login sur l'application, et je vois encore que mon adresse émail est en texte clair lors de l'intcept du traffic :

Burp Suite Professional v2022.11.4 - Temporary Project - licensed to surfonyz

Request Attributes: 2

Request Query Parameters: 1

Request Body Parameters: 0

Request Cookies: 0

Request Headers: 6

Inspector

HTTP/2

Comment this item

Forward Drop Intercept is on Action Open Browser

Pretty Raw Hex

```

1 POST /api/nordpass/providers?email=ismail.arame66@gmail.com HTTP/2
2 Host: api.nordpass.com
3 User-Agent: NordPass/Android (playstore/4.2.25 40225) Android 12
4 Accept-Encoding: gzip, deflate
5
6

```

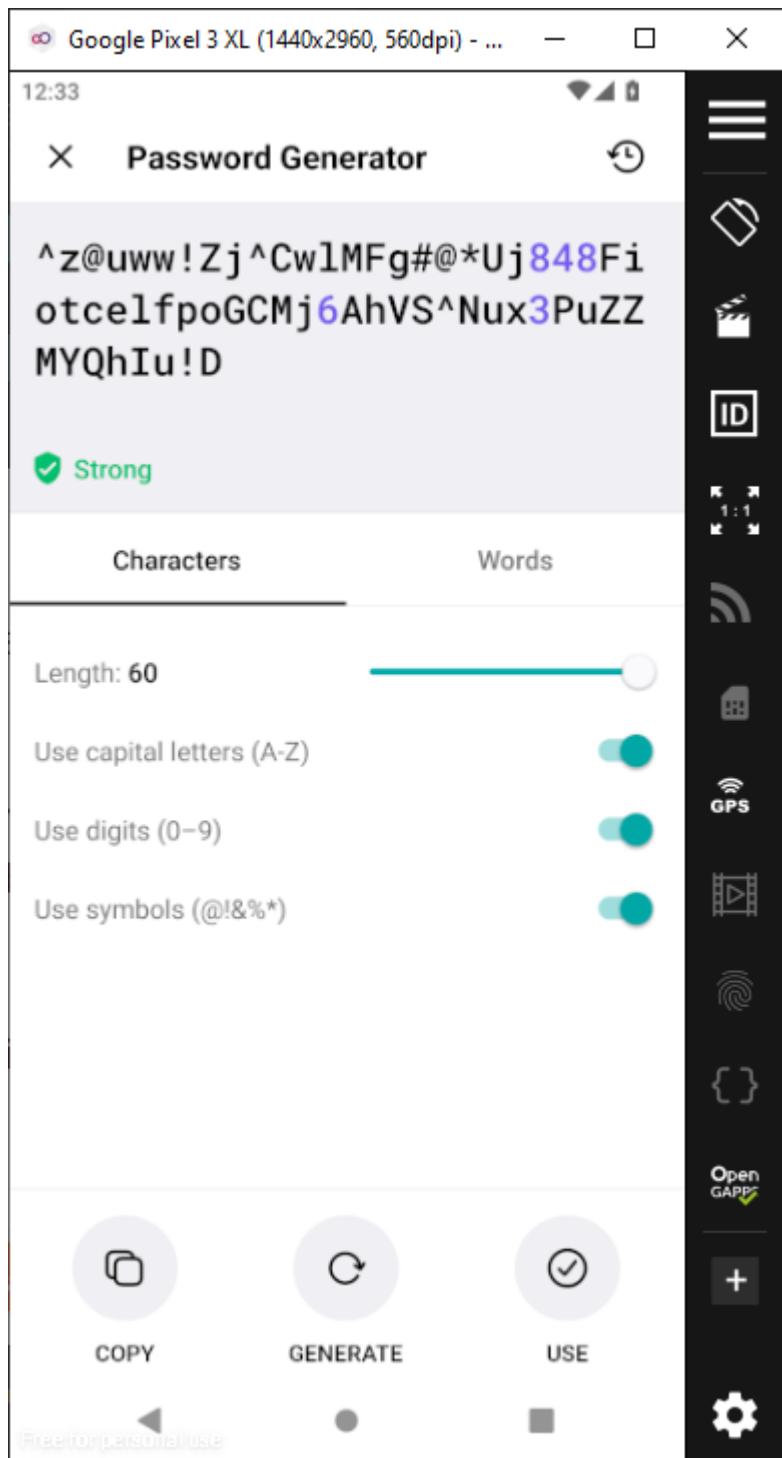
- ✓ Ensuite NordPass me demande de créer un master password et le confirmer, et puis l'application m'a assigné un id et des clés publiques et privées chiffrées :

The screenshot shows a Burp Suite interface with a captured POST request to `https://api.nordpass.com:443`. The request payload is a JSON object containing various fields such as `identity_id`, `keys`, `dek_info`, `encrypted_private_key`, `change_editor_signature`, `change_password`, `item_id`, `item_key`, and `item_signature`. The response is a mobile application dialog titled "Confirm Master Password" with the message "It is important to remember it! Make sure you have got it right." and the email "ismail.arame66@gmail.com - Log Out".

✓ D'autre, NordPass offre la possibilité de stocker les mots de passe des différentes applications. En titre d'exemple j'ai ajouté le mot de passe de mon compte facebook et aucune des informations que j'ai saisi est fachée, le trafic est chiffré avec des clés encore chiffrées :

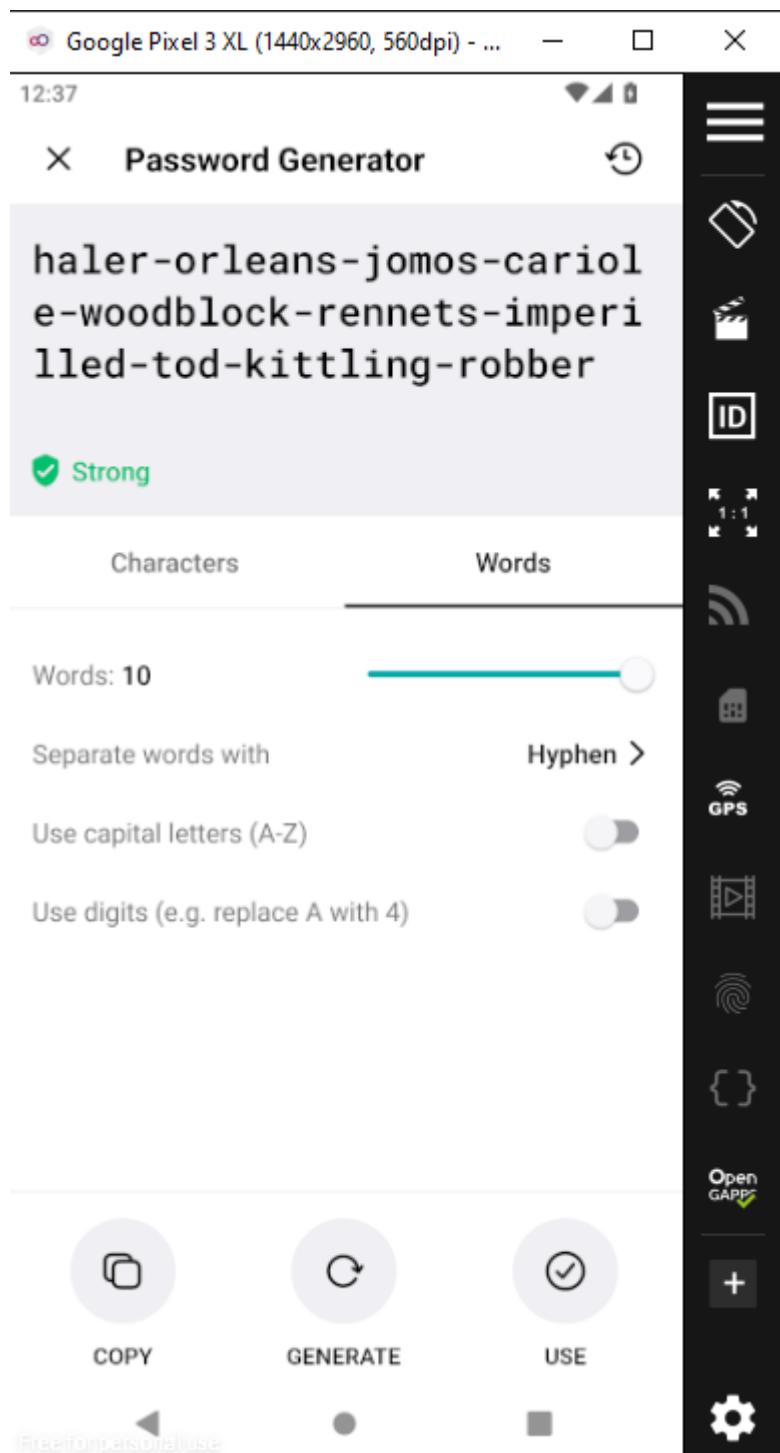
The screenshot shows a Burp Suite interface with a captured POST request to `https://api.nordpass.com:443`. The request payload is a JSON object with fields like `change`, `change_editor_signature`, `change_password`, `item_id`, `item_key`, and `item_signature`. The response is a mobile application dialog titled "Add Password" for Facebook, showing fields for "Title" (Facebook), "Email or Username" (ismail.arame@gmail.com), "Password" (sayonara129), and "Websites and Apps" section with "Website or App Name" (https://facebook.com/).

- Pour maintenir une sécurité de mot de passe uncrackable, NordPass fournit des générateurs de mots de passe robustes accessibles à tous, même sans abonnement.
- **NordPass** facilite la création de mots de passe hautement sécurisés d'une longueur maximale de 60 caractères, comprenant un mélange de lettres majuscules et minuscules, de chiffres et de symboles spéciaux.



- ✓ Il se distingue notamment par sa fonction de générateur de phrases de passe, permettant aux utilisateurs de créer des phrases de passe allant jusqu'à 10 mots, offrant une solution à

la fois sécurisée et plus facile à mémoriser par rapport à une série aléatoire de caractères.



➤ Keeper :



✓ Android.manifest.xml -

il y en a une seul activite qui peut etre execute par d'autre app.

```
AndroidManifest.xml x
Find: exported="true"
249      <category android:name="android.intent.category.DEFAULT"/>
248  </intent-filter>
243 </activity>
252 <activity android:theme="@style/ImportPasswords" android:name="com.callpod.android_apps.keeper.onboarding.ImportPasswor
258   <intent-filter>
259     <category android:name="android.intent.category.DEFAULT"/>
258   </intent-filter>
252 </activity>
262 <activity android:theme="@style/Theme.Keeper.Light" android:label="@string/app_name" android:name="com.callpod.android_
268 <activity android:theme="@style/Theme.Keeper" android:name="com.callpod.android_apps.keeper.DetailLoadActivity" android
274   <intent-filter>
275     <action android:name="com.callpod.android_apps.keeper.DetailLoadActivity"/>
277   <category android:name="android.intent.category.DEFAULT"/>
274 </intent-filter>
268 </activity>
280 <activity android:theme="@android:style/Theme.NoDisplay" android:label="@string/app_name" android:name="com.callpod.and
284 <activity android:name="com.callpod.android_apps.keeper.ParseDeepLinkActivity" android:exported="true" android:launche
288   <intent-filter>
289     <action android:name="android.intent.action.VIEW"/>
291     <category android:name="android.intent.category.DEFAULT"/>
292     <category android:name="android.intent.category.BROWSABLE"/>
294     <data android:scheme="https" android:host="@string/deeplinkDomain" android:pathPrefix="/buy"/>
288 </intent-filter>
299 <intent-filter>
300   <action android:name="android.intent.action.VIEW"/>
302   <category android:name="android.intent.category.DEFAULT"/>
303   <category android:name="android.intent.category.BROWSABLE"/>
305   <data android:scheme="https" android:host="@string/deeplinkDomain" android:pathPrefix="/vault"/>
309   <data android:scheme="https" android:host="@string/euDeepLinkDomain" android:pathPrefix="/vault"/>
299 </intent-filter>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="572" android:versionName="15.0.1.1" and
<uses-sdk android:minSdkVersion="23" android:targetSdkVersion="29"/>
<supports-screens android:anyDensity="true" android:smallScreens="true" android:normalScreens="true" android:largeScreens="tr
<uses-feature android:name="android.hardware.wifi" android:required="false"/>
<uses-feature android:name="android.hardware.telephony" android:required="false"/>
<uses-feature android:name="android.hardware.screen.portrait" android:required="false"/>
<uses-feature android:name="android.hardware.touchscreen"/>
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE"/>
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED"/>
<uses-permission android:name="android.permission.VIBRATE"/>
<uses-permission android:name="android.permission.USE_BIOMETRIC"/>
<uses-permission android:name="android.permission.FOREGROUND_SERVICE"/>
<uses-permission android:name="com.android.vending.BILLING"/>
<uses-permission android:name="android.permission.READ_CONTACTS"/>
<uses-permission android:name="android.permission.GET_ACCOUNTS"/>
<uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
<uses-permission android:name="android.permission.RECORD_AUDIO"/>
<uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW"/>
<uses-permission android:name="android.permission.CAMERA"/>
<uses-permission android:name="android.permission.WAKE_LOCK"/>
<uses-permission android:name="com.google.android.c2dm.permission.RECEIVE"/>
<uses-permission android:name="android.permission.USE_FINGERPRINT"/>
<uses-permission android:name="com.google.android.finsky.permission.BIND_GET_INSTALL_REFERRER_SERVICE"/>
<uses-feature android:name="android.hardware.camera" android:required="false"/>
<uses-feature android:name="android.hardware.camera.autofocus" android:required="false"/>
<application android:theme="@style/Theme.Keeper" android:label="Keeper" android:icon="@mipmap/ic_launcher" android:name="com.
  <meta-data android:name="com.google.android.gms.vision.DEPENDENCIES" android:value="barcode"/>
  <activity android:theme="@style/Theme.Keeper.Launcher" android:name="com.callpod.android_apps.keeper.ResultsActivity" and
    <intent-filter>
```

✓ Resources.arc/res/values/strings:

On a trouve ce cle API :

```
34 <string name="get_verification_code_button">Send Code</string>
35 <string name="good_email_package">com.good.android.gfe</string>
36 <string name="google_api_key">AIzaSyCkTmHw3kL1rxdsQN71LinE24Vpmf2dWuI</string>
37 <string name="google_app_id">1:36874242427:android:701c34fe643c0444</string>
38 <string name="google_authenticator">Google and Microsoft Authenticator (TOTP)</string>
39 <string name="google_crash_reporting_api_key">AIzaSyCkTmHw3kL1rxdsQN71LinE24Vpmf2dWuI</string>
40 <string name="google_storage_bucket">graphite-wave-684.appspot.com</string>
41 <string name="got_it">Got It</string>
```

✓ URLs:

✓ Analyse du code :

dans com.callpod.android_apps.keeper.common.password

- on trouve la fonction `generateRandomizedPassword()`.

```
private final String generateRandomizedPassword() {
    PasswordConfig passwordConfig = this.config;
    if (passwordConfig == null) {
        Intrinsics.throwUninitializedPropertyAccessException("config");
    }
    int max = Math.max(Math.min(passwordConfig.getLength(), 200), 4);
    PasswordConfig passwordConfig2 = this.config;
    if (passwordConfig2 == null) {
        Intrinsics.throwUninitializedPropertyAccessException("config");
    }
    int useDigitsMin = passwordConfig2.getUseDigitsMin();
    PasswordConfig passwordConfig3 = this.config;
    if (passwordConfig3 == null) {
        Intrinsics.throwUninitializedPropertyAccessException("config");
    }
    int useUpperCaseMin = useDigitsMin + passwordConfig3.getUseUpperCaseMin();
    PasswordConfig passwordConfig4 = this.config;
    if (passwordConfig4 == null) {
        Intrinsics.throwUninitializedPropertyAccessException("config");
    }
    int useSymbolsMin = useUpperCaseMin + passwordConfig4.getUseSymbolsMin();
    PasswordConfig passwordConfig5 = this.config;
    if (passwordConfig5 == null) {
```

- path: androidx.security.crypto

```

/* Loaded from: classes.dex */
public final class MasterKeys {
    private static final String ANDROID_KEYSTORE = "AndroidKeystore";
    static final String KEYSTORE_PATH_URI = "android-keystore://";
    private static final int KEY_SIZE = 256;
    static final String MASTER_KEY_ALIAS = "_androidx_security_master_key_";
    public static final KeyGenParameterSpec AES256_GCM_SPEC = createAES256GCMKeyGenParameterSpec(MASTER_KEY_ALIAS);

    private MasterKeys() {
    }

    private static KeyGenParameterSpec createAES256GCMKeyGenParameterSpec(String str) {
        return new KeyGenParameterSpec.Builder(str, 3).setBlockModes(CodeGenPackage.GCM).setEncryptionPaddings("NoPadding").setKeySize(256).build();
    }

    public static String getOrCreate(KeyGenParameterSpec keyGenParameterSpec) throws GeneralSecurityException, IOException {
        validate(keyGenParameterSpec);
        if (!keyExists(keyGenParameterSpec.getKeyStoreAlias())) {
            generateKey(keyGenParameterSpec);
        }
        return keyGenParameterSpec.getKeyStoreAlias();
    }

    static void validate(KeyGenParameterSpec keyGenParameterSpec) {
        if (keyGenParameterSpec.getKeySize() != 256) {
            throw new IllegalArgumentException("invalid key size, want 256 bits got " + keyGenParameterSpec.getKeySize() + " bits");
        } else if (!Arrays.equals(keyGenParameterSpec.getBlockModes(), new String[]{CodeGenPackage.GCM})) {
            throw new IllegalArgumentException("invalid block mode, want GCM got " + Arrays.toString(keyGenParameterSpec.getBlockModes()));
        } else if (keyGenParameterSpec.getPurposes().length != 3) {
            throw new IllegalArgumentException("invalid purposes mode, want PURPOSE_ENCRYPT | PURPOSE_DECRYPT got " + keyGenParameterSpec.getPurposes());
        } else if (!Arrays.equals(keyGenParameterSpec.getEncryptionPaddings(), new String[]{"NoPadding"})) {
            throw new IllegalArgumentException("invalid padding mode, want NoPadding got " + Arrays.toString(keyGenParameterSpec.getEncryptionPaddings()));
        } else if (keyGenParameterSpec.isUserAuthenticationRequired() && keyGenParameterSpec.getUserAuthenticationValidityDurationSeconds() < 1) {
            throw new IllegalArgumentException("per-operation authentication is not supported (UserAuthenticationValidityDurationSeconds must be >0)");
        }
    }

    private static void generateKey(KeyGenParameterSpec keyGenParameterSpec) throws GeneralSecurityException {
        KeyGenerator keyGenerator = KeyGenerator.getInstance("AES", ANDROID_KEYSTORE);
        keyGenerator.init(keyGenParameterSpec);
        keyGenerator.generateKey();
    }
}

```

Dynamic analysis :

Proxy Listeners

Burp Proxy uses listeners to receive incoming HTTP requests from your browser. You will need to configure your browser to use one of the listeners as its proxy server.

	Running	Interface	Invisible	Redirect	Certificate	TLS Protocols
Add						
Edit	<input checked="" type="checkbox"/> 127.0.0.1:8080				Per-host	Default
Remove	<input checked="" type="checkbox"/> *:8082				per-host	Default

CA Certificate

Choose a file to export the CA certificate.

C:\Users\rando\OneDrive\Bureau\Burp_CA.CER Select file ...

Import / export CA certificate Regenerate CA certificate

Intercept Client Requests

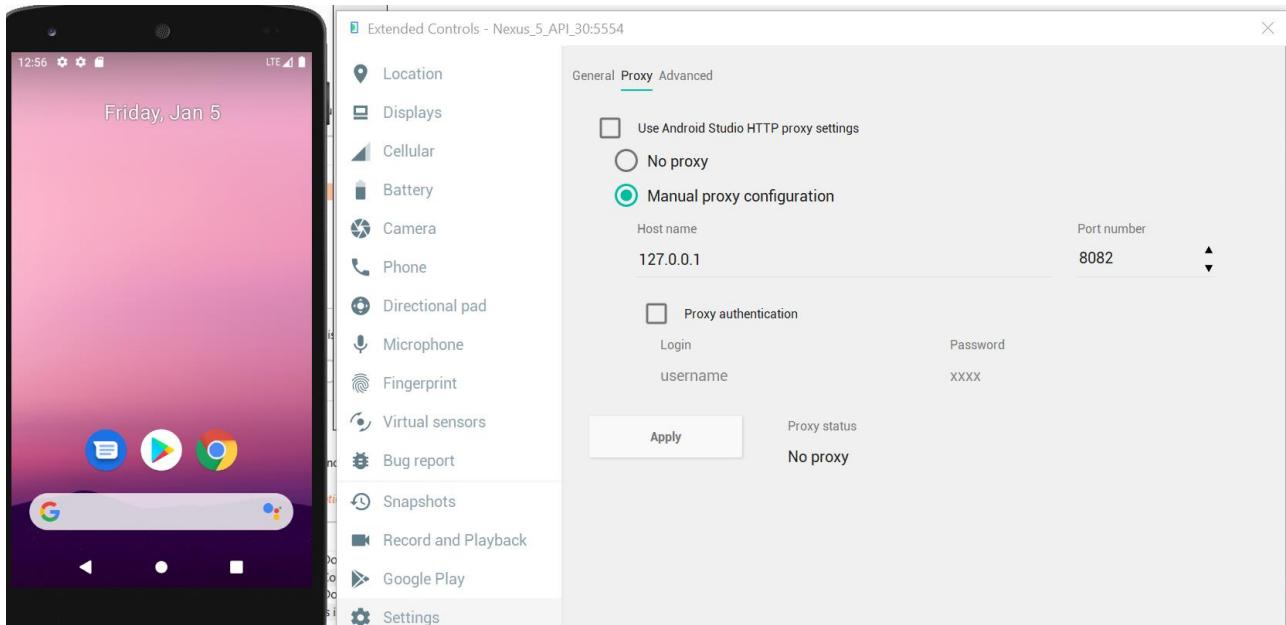
Use these settings to control which requests are stalled for viewing and editing in the Intercept tab.

Intercept requests based on the following rules: *Master interception is turned off*

Enabled	Operator	Match type	Relationship	Condition
<input checked="" type="checkbox"/>	File extension	Does not match	(^gif\$ ^jpg\$ ^png\$ ^css\$ ^js\$ ^ico\$)	
<input type="checkbox"/>	Or	Request	Contains parameters	
<input type="checkbox"/>	Or	HTTP method	Does not match	(get post)
<input type="checkbox"/>	And	URL	Is in target scope	

Back Next

✓ Créer un listener



✓ Injection et execution de frida-server:

```
C:\Users\rando\OneDrive\Bureau>py -m pip install frida
Collecting frida
  Downloading frida-16.1.10-cp37-abi3-win_amd64.whl (32.6 MB)
    32.6/32.6 MB 438.7 kB/s eta 0:00:00
Installing collected packages: frida
Successfully installed frida-16.1.10
[notice] A new release of pip available: 22.3.1 >= 23.3.2
[notice] To update, run: C:\Users\rando\AppData\Local\Programs\Python\Python311\python.exe -m pip install --upgrade pip
C:\Users\rando\OneDrive\Bureau>py -m pip install objection
Collecting objection
  Downloading objection-1.11.0.tar.gz (327 kB)
    327.2/327.2 kB 53.1 kB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: frida>=14.0.0 in c:\users\rando\appdata\local\programs\python\python311\lib\site-packages (from objection)
Collecting frida-tools>=6.0.0
  Downloading frida-tools-12.3.0.tar.gz (200 kB)
    200.5/200.5 kB 81.2 kB/s eta 0:00:00
  Installing build dependencies ... done
  Getting requirements to build wheel ... done
  Preparing metadata (pyproject.toml) ... done
Collecting prompt_toolkit<4.0.0,>=3.0.3
  Downloading prompt_toolkit-3.0.43-py3-none-any.whl (386 kB)
    386.1/386.1 kB 250.6 kB/s eta 0:00:00

```

```
C:\Users\rando\Downloads>adb push "C:\Users\rando\Downloads\frida-server-16.1.10-android-x86\frida-server-16.1.10-android-x86" data/local/tmp
C:\Users\rando\Downloads\frida-server-16.1.10-android-x86\frida-server-16.1.10-android-x86: 1 file pushed, 0 skipped, 16.8 MB/s (51684860 bytes in 2.926s)

C:\Users\rando\Downloads>adb shell "chmod +x data/local/tmp/frida-server-16.1.10-android-x86"
C:\Users\rando\Downloads>adb shell "data/local/tmp/frida-server-16.1.10-android-x86 &"
```

✓ Bypass SSL pinning

✓ Voila le traffic apres le SSL pinning bypass

The image shows a mobile phone screen with a 'Create Account' form overlaid on a browser window. The browser window at the top has tabs for 'Intercept', 'HTTP history', 'WebSockets history', and 'Options'. Below the tabs are buttons for 'Forward', 'Drop', 'Intercept is on' (which is highlighted in blue), 'Action', and 'Open Browser'. The main content area shows a POST request to '/api/test/authentication/request_create_user' with the following details:

```
POST /api/test/authentication/request_create_user HTTP/2
Host: keepersecurity.com
Content-Type: application/octet-stream
Content-Length: 243
Accept-Encoding: gzip, deflate
User-Agent: Callipod Keeper for Android 1.0 (15.0.1.1/572) Dalvik/2.1.0 (Linux; U; Android 7.0; Android SDK built for x86 Build/NYC)

```

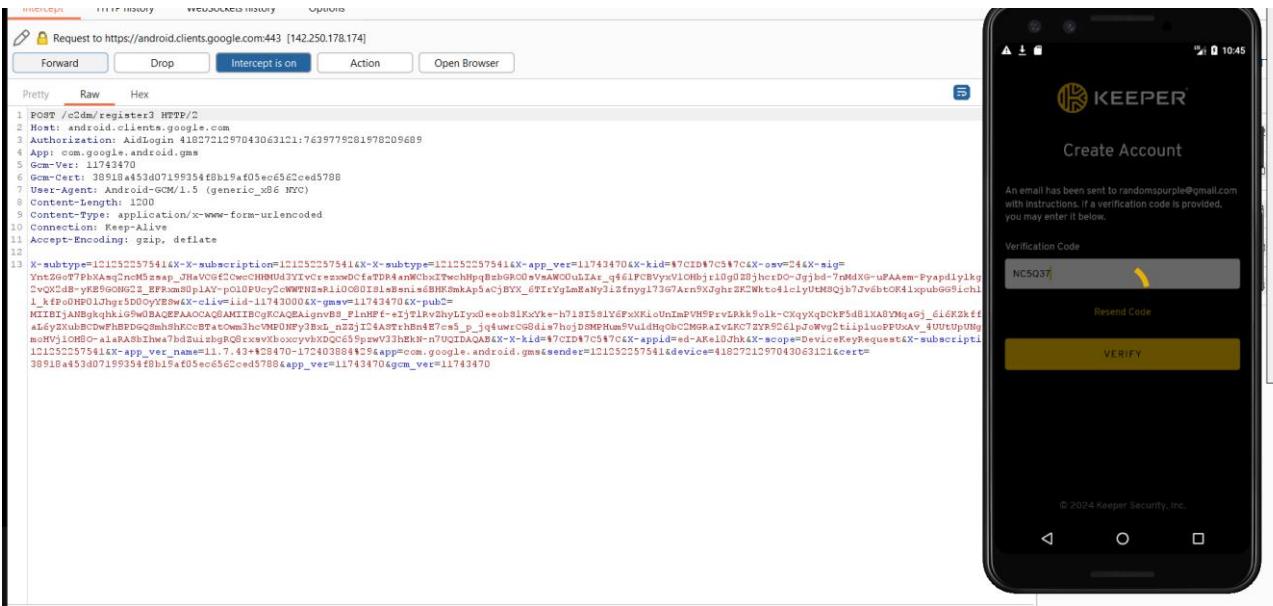
The body of the POST request contains a large amount of encoded data, likely a JSON object, which includes fields such as 'Email' (randompurple@gmail.com), 'Master Password' (a masked password), 'Confirm Master Password' (a masked password), and 'Next' (a button labeled 'NEXT'). The browser's status bar at the bottom right shows the time as 10:39.

- Deux requêtes peuvent être exploitées par un attaquant (**MITM attack**) car les données ne sont pas chiffrées:

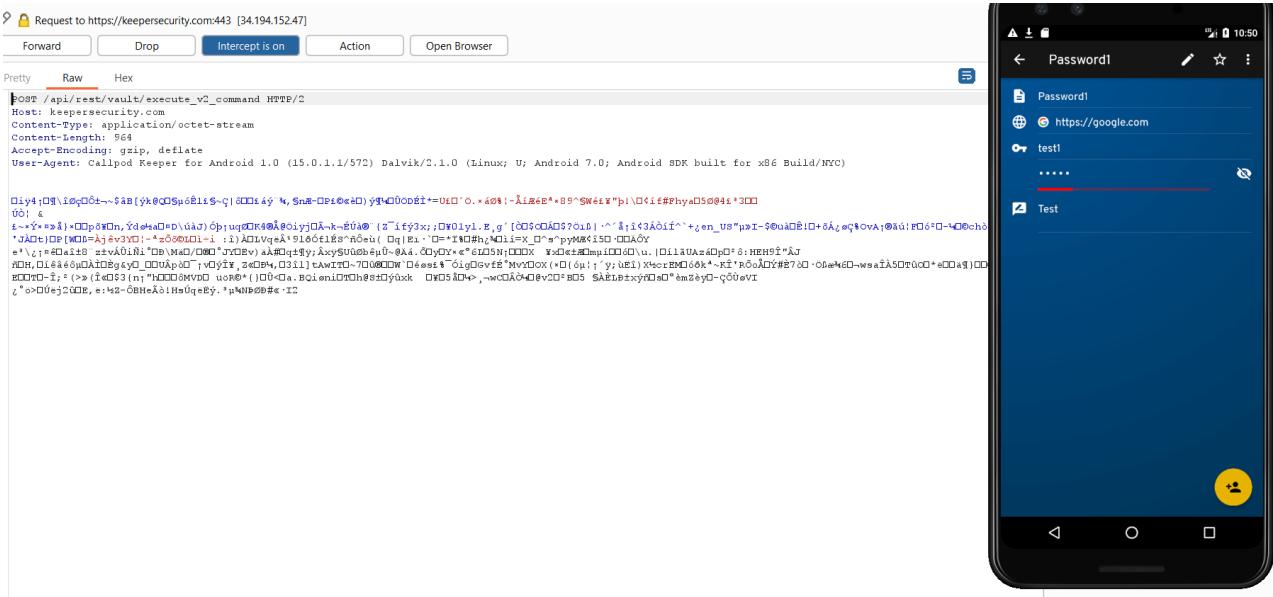
The screenshot shows a mobile application interface for Keeper Security. At the top, there's a navigation bar with icons for back, forward, and search, followed by buttons for "Forward", "Drop", "Intercept is on" (which is currently selected), "Action", and "Open Browser". Below this is a toolbar with "Pretty", "Raw", and "Hex" options, along with a copy icon. The main content area displays a POST request to https://keepersecurity.com:443. The request body is a JSON object containing various parameters:

```
1 POST /emergency_check HTTP/2
2 Host: keepersecurity.com
3 Content-Type: application/x-www-form-urlencoded
4 Content-Length: 505
5 Accept-Encoding: gzip, deflate
6 User-Agent: Calipod Keeper for Android 1.0 (15.0.1.1/572) Dalvik/2.1.0 (Linux; U; Android 7.0; Android SDK built for x86 Build/NYC)
7
8 appstore_type=pid=244carrier=Android&email=randompurple4@gmail.com&adjust_client_id=763hbC4c-ahlc-44Ce-bac-3-Bed471546a19cmfg=Google&build_type=gplayProduction&format=json&code=fv=15.0.1&adjust_aid=&country=US&device=generic_x86&session_token=0U9c1pJnDongkqy0ALInlkVg_MmMHNfLXNlMD-2k2mIoBsuVNQm2-WMCB4FuBjzCnW4ejr5ZKSL1DK3MfaQXctheime=CLASfC_BDUt4&brand=google&recs=0&appstore_token=&uid=W8gJwvJdCY7-kxy9ZYaqAlanguage=en&mcu_mnc=3102&product=sdk_google_phone_x86&model=Android&SDK=built+for+x86
```

To the right of the request, there's a "Create Account" button and a note: "An email has been sent to randompurple@gmail.com with instructions. If a verification code is provided, you may enter it below." Below this is a "Verification Code" input field containing "NC5Q37" with a yellow pencil icon, and a "VERIFY" button.



j'ai essayé d'ajouter un website sur l'application



➤ mSecure :

⊕ Static analysis :

✓ Androidmanifest.xml:

il y en a un seul **content provider** qui peut etre utiliser par les autres app

```
<activity android:name=".com.mseven.msecure.ItemInfo" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.NewField" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.PasswordValidation" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale" android:windowSoftInputMode="stateAlwaysVisible"/>
<activity android:name=".com.mseven.msecure.Login" android:configChanges="keyboardHidden|keyboard|locale" android:windowSoftInputMode="stateAlwaysVisible"/>
<activity android:name=".com.mseven.msecure.About" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.PasswordGenerator" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<receiver android:name=".com.mseven.msecure.BootService">
    <intent-filter>
        <action android:name="android.intent.action.BOOT_COMPLETED"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</receiver>
<activity android:name=".com.mseven.msecure.ReorderFields" android:configChanges="screenSize|orientation|keyboardHidden|locale"/>
<activity android:name=".com.mseven.msecure.EditConnection" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.SettingsConnection" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.GmailSetting" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.SendBackupEmail" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.SyncPassword" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.GetSupport" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.CopyToClipboard" android:launchMode="singleInstance" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.mBrowser"/>
<activity android:name=".com.mseven.msecure.mSecureTabs" android:launchMode="singleTask" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.ForcedOverflowItem" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.dropbox.client2.android.AuthActivity" android:launchMode="singleTask" android:configChanges="screenSize|orientation|keyboard|locale">
    <intent-filter>
        <data android:scheme="db-mhhiubxwnfhv0"/>
        <action android:name="android.intent.action.VIEW"/>
        <category android:name="android.intent.category.BROWSABLE"/>
        <category android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
<activity android:name=".com.mseven.msecure.WelcomeScreen" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.CropImage.CropImage"/>
<activity android:name=".com.mseven.msecure.RecoverDeletedItems" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<activity android:name=".com.mseven.msecure.DisplayImage" android:configChanges="screenSize|orientation|keyboardHidden|keyboard|locale"/>
<provider android:name="InternalStorageContentProvider" android:exported="true" android:authorities=".com.mseven.msecure"/>
<activity android:name=".a.y.z.A" android:configChanges="screenSize|screenLayout|orientation|keyboardHidden"/>
<meta-data android:name="default-activity" android:value=".com.mseven.msecure.Licensing"/>
<activity android:name=".a.y.z.HZ" android:configChanges="screenSize|screenLayout|orientation|keyboardHidden|keyboard|locale" android:windowSoftInputMode="stateAlwaysHidden">
    <intent-filter>
        <category android:name="android.intent.category.LAUNCHER"/>
        <action android:name="android.intent.action.MAIN"/>
    </intent-filter>
</activity>
</application>
...
```

✓ META-INF/MANIFEST.MF :

```
1 Manifest-Version: 1.0
2 Created-By: 1.8.0_45 (Oracle Corporation)
3
4 Name: res/drawable-xhdpi-v4/abs__progress_secondary_holo_dark.9.png
5 SHA1-Digest: D0QKteBrrHhkWE9BB+d8/oInB=
6
7 Name: res/drawable-xxhdpi-v4/ic_action_add.png
8 SHA1-Digest: KJMeq5pZx/JncZi0JPmXcdcrkD8=
9
10 Name: res/drawable/popup_side_img.png
11 SHA1-Digest: CsIBflPh3MNs5Qjniz6+r3zyLYU=
12
13 Name: res/drawable-mdpi-v4/abs_ab_solid_light_holo.9.png
14 SHA1-Digest: qPBmGe92FV9luR4XdeDl18Xokio=
15
16 Name: res/drawable/add_tab.png
17 SHA1-Digest: JfmbDUdV3PL5Id83v14ftxWf6fA=
18
19 Name: res/drawable-ndpi-v4/music2.png
20 SHA1-Digest: 2P9hoxDufi6X8dS8ix+9et+q1g=
21
22 Name: res/drawable-ndpi-v4/note.png
23 SHA1-Digest: TrQ4nxnfngqOs32WRdLU4Be2cNPc=
24
25 Name: res/layout/cropimage.xml
26 SHA1-Digest: bFbQip7Ld50ZEKKX/Q47U33jHR2A=
27
28 Name: res/drawable-xhdpi-v4/ic_menu_settings_white.png
29 SHA1-Digest: aokhNtwCTsOBVa3Rm0/4W3kTbt=
30
31 Name: res/drawable/menu_trash.png
32 SHA1-Digest: W62qzQu/qwnkeIK1D13KJu0n/18=
33
34 Name: res/layout/editgroup.xml
35 SHA1-Digest: urHj4VKto5WKhbgjRZX4u642HA=
36
37 Name: res/drawable-ndpi-v4/web_violet.png
38 SHA1-Digest: BH/H9Yk90HnUQ0mlGVJ8cy7Cis=
39
40 Name: res/drawable/customdrawablecheckbox.xml
41 SHA1-Digest: P2UZIVRa+L00h5e/Pihr7pGiobc=
42
43 Name: res/drawable-ndpi-v4/burger_template.png
44 SHA1-Digest: vvwIRkn5Fk7inOhTc43H/7rRPVvk=
```

✓ Res/values/strings.xml:

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3     <string name="abs__action_bar_home_description">Navigate home</string>
4     <string name="abs__action_bar_up_description">Navigate up</string>
5     <string name="abs__action_menu_overflow_description">More options</string>
6     <string name="abs__action_mode_done">Done</string>
7     <string name="abs__activity_chooser_view_see_all">See all...</string>
8     <string name="abs__activity_chooser_view_dialog_title_default">Select activity</string>
9     <string name="abs__share_action_provider_share_with">Share with...</string>
10    <string name="abs__activitychooserview_choose_application">Choose an application</string>
11    <string name="abs__shareactionprovider_share_with">Share with</string>
12    <string name="abs__shareactionprovider_share_with_application">Share with %s</string>
13    <string name="dig_msg_payment_success">The payment was successful</string>
14    <string name="IDS_SAPPS_POP_AN_ERROR_OCCURRED_WHILE_RESETTING_SAMSUNG_IN_APP_PURCHASE">An error occurred while resetting Samsung In-App Purchase</string>
15    <string name="IDS_SAPPS_POP_AN_INVALID_VALUE_HAS_BEEN_PROVIDED_FOR_SAMSUNG_IN_APP_PURCHASE">An invalid value has been provided for Samsung In-App Purchase</string>
16    <string name="IDS_SAPPS_POP_AN_UNEXPECTED_ERROR_HAS_OCCURRED_SAMSUNG_ACCOUNT_AUTHENTICATION_HAS_BEEN_CANCELLED">An unexpected error has occurred. Samsung account authentication has been cancelled</string>
17    <string name="IDS_SAPPS_POP_YOUR_PURCHASE_VIA_SAMSUNG_IN_APP_PURCHASE_IS_INVALID_A_FAKE_APPLICATION_HAS BEEN_DETECTED_CHECK_THE_APP_MSG">Your purchase via Samsung In-App Purchase is invalid. A fake application has been detected. Check the app message</string>
18    <string name="IDS_SAPPS_POP_UNKNOVN_ERROR_OCCURRED">Unknown error occurred</string>
19    <string name="IDS_SAPPS_POP_SAMSUNG_IN_APP_PURCHASE">Samsung In-App Purchase</string>
20    <string name="IDS_SAPPS_POP_SAMSUNG_IN_APP_PURCHASE_NEEDS_TO_BE_UPDATED">Samsung In-App Purchase needs to be updated</string>
21    <string name="IDS_SAPPS_POP_TO_PURCHASE_ITEMS_YOU_NEED_TO_INSTALL_SAMSUNG_IN_APP_PURCHASE_INSTALL_Q">To purchase items, you need to install Samsung In-App Purchase. Install Samsung In-App Purchase</string>
22    <string name="IDS_SAPPS_POP_A_NEW_VERSION_OF_SAMSUNG_IN_APP_PURCHASE_IS_AVAILABLE_UPDATE_Q">A new version of Samsung In-App Purchase is available. Update?</string>
23    <string name="IDS_SAPPS_BODY_WAITING_ING">Waiting...</string>
24    <string name="IDS_SAPPS_POP_PRODUCT_DOES_NOT_EXIST_IN_THIS_STORE">Product does not exist in this store</string>
25    <string name="IDS_SAPPS_POP_ALREADY_PURCHASED">Already purchased</string>
26    <string name="IDS_SAPPS_POP_PAYMENT_CANCELLED">Payment cancelled</string>
27    <string name="IDS_SAPPS_POP_NETWORK_UNAVAILABLE">Network not available</string>
28    <string name="IDS_SAPPS_HEADER_ITEM_PURCHASED">Item purchased</string>
29    <string name="delQntT">Are you sure you want to delete this type?</string>
30    <string name="errorStr">Error</string>
31    <string name="cancelStr">Cancel</string>
32    <string name="GmailCancelBttnStr">Cancel</string>
33    <string name="passgenCancelText">Cancel</string>
34    <string name="SetPassCancelBttnStr">Cancel</string>
35    <string name="syncAlertCancelStr">Cancel</string>
36    <string name="syncCancelBut5t">Cancel</string>
37    <string name="SyncEmailSyncOrRestoreDialogCancel1ButtonStr">Cancel</string>
38    <string name="WIFINotConnectedDialogCancel1BtnStr">Cancel</string>
39    <string name="cancelEditStr">Cancel Edit type</string>
40    <string name="ProcessCancelledStr">Canceling operation, please wait ...</string>
41    <string name="fontSizeStr">Change Font Size</string>
42    <string name="changePasswordSummaryStr">Change or clear the password used in mSecure</string>
43    <string name="changePasswordStr">Change Password</string>
44    <string name="CheckingDropBoxExist">Checking for Dropbox sync file</string>
45    <string name="recommendEmailSubjectStr">Checkout mSecure Password Manager</string>
46    <string name="recommendBodyStr">Checkout mSecure Password Manager for Android, iOS, Windows and MacOS\nhttp://msevensoftware.com</string>
```

✓ URL:

Search for text: Auto search

Search definitions of: Class Method Field Code Resource Comments

Search options: Case-insensitive Regex Active tab only

```
<public type="layout" name="http_authentication" id="0x7f030038" />
<string name="recommendBodyStr">歡迎瞭解用於Android、iOS、Windows和MacOS系統的mSecure密碼助手。http://msevensoftware.com</string>
<string name="recommendBodyStr">歡迎瞭解用於Android、iOS、Windows和MacOS系統的mSecure密碼助手。http://msevensoftware.com</string>
<string name="recommendBodyStr">Android、iOS、Windows 和 MacOS 和 mSecure 開源 軟件 http://msevensoftware.com</string>
<string name="recommendBodyStr">Android、iOS、Windows、そしてMac用のOmSecureパスワードマネージャーを確認してください。http://msevensoftware.com</string>
<string name="recommendBodyStr">Оформить заказ на Менеджер Паролей mSecure для Android, iOS, Windows и MacOS http://msevensoftware.com</string>
<string name="recommendBodyStr">Verifique o seu gerenciador de senhas mSecure para Android, iOS, Windows e MacOS http://msevensoftware.com</string>
<string name="recommendBodyStr">Verifique el administrador de contraseñas de mSecure para Android, iOS, Windows y MacOS http://msevensoftware.com</string>
<string name="recommendBodyStr">Kasa mSecure Password Manager dla Android, iOS, Windows i MacOS http://msevensoftware.com</string>
<string name="recommendBodyStr">Verifica Gestore password di mSecure per Android, iOS, Windows e MacOS http://msevensoftware.com</string>
<string name="recommendBodyStr">Commandez le Gestionnaire de mot de passe mSecure pour Android, iOS, Windows et MacOS http://msevensoftware.com</string>
<string name="about_privacy_web_str">http://msevensoftware.com/privacy</string>
<string name="about_permissions_web_str">http://msevensoftware.com/android\_users\_guide#Permissions</string>
<string name="about_help_web_str">>http://msevensoftware.com/android\_users\_guide</string>
<string name="about_privacy_web_str">>http://msevensoftware.com/privacy</string>
<string name="about_permissions_web_str">>http://msevensoftware.com/android\_users\_guide#Permissions</string>
<string name="about_help_web_str">>http://msevensoftware.com/android\_users\_guide</string>
<string name="about_privacy_web_str">>http://msevensoftware.com/privacy</string>
<string name="about_permissions_web_str">>http://msevensoftware.com/android\_users\_guide#Permissions</string>
<string name="about_help_web_str">>http://msevensoftware.com/android\_users\_guide</string>
<string name="about_privacy_web_str">>http://msevensoftware.com/privacy</string>
```

Found 50+

Dynamique Analysis :

- ✓ On lance ssl pinning certificate bypass comme l'exemple précédent mais maintenant pour l'application mSecure

```
Administrator: Command Prompt - frida --codeshare pcipolloni/universal-android-ssl-pinning-bypass-with-frida -U -f com.mseven.barolo

C:\Users\azer\Documents\burp>frida --codeshare pcipolloni/universal-android-ssl-pinning-bypass-with-frida -U -f com.mseven.barolo

    / _ |  Frida 16.1.7 - A world-class dynamic instrumentation toolkit
   | ( ) |
 > _ |  Commands:
 / /_ |  help      -> Displays the help system
 . . . |  object?   -> Display information about 'object'
 . . . |  exit/quit -> Exit
 . . . |
 . . . More info at https://frida.re/docs/home/
 . . . |
 . . . Connected to Pixel 3 XL (id=192.168.60.101:5555)
Spawned 'com.mseven.barolo'. Resuming main thread!
[Pixel 3 XL::com.mseven.barolo ]->
[.] Cert Pinning Bypass/Re-Pinning
[+] Loading our CA...
[o] Our CA Info: CN=PortSwigger CA, OU=PortSwigger CA, O=PortSwigger, L=PortSwigger, ST=PortSwigger, C=PortSwigger
[+] Creating a KeyStore for our CA...
[+] Creating a TrustManager that trusts the CA in our KeyStore...
[+] Our TrustManager is ready...
[+] Hijacking SSLContext methods now...
[-] Waiting for the app to invoke SSLContext.init()...
[o] App invoked javax.net.ssl.SSLContext.init...
[+] SSLContext initialized with our custom TrustManager!
```

- ✓ J'ai commencé après par créer un compte sur mSecure, l'application fait un check du password simultanément en envoyant une requette checkPasswordStrength vers le server mSecure :

The screenshot shows the Burp Suite Professional interface. On the left, the Proxy tab is selected, showing an intercept for a POST request to `https://msecure-docker-parse.herokuapp.com:443`. The request payload is a JSON object with a single key-value pair: `{"password": "ca45"}`. On the right, the application's sign-up screen is displayed. It has fields for Email (set to `ismail.arame66@gmail.com`) and Password (set to `8845`). Below the password field, there is a note: `IMPORTANT: Your account password will be used to unlock mSecure.` At the bottom, there are `ACCOUNT HELP?` and `CONTINUE` buttons.

- ✓ après l'application fait un check pour vérifier la duplication de email c.a.d s'il est déjà enregistré et s'il est unique en utilisant la fonction checkDuplicateUsername

Burp Suite Professional v2022.11.4 - Temporary Project - licensed to surfenyz

Request to https://msecure-docker-parse.herokuapp.com:443 [3.209.172.72]

```

1 POST /parse/functions/checkDuplicateUsername HTTP/1.1
2 Host: msecure-docker-parse.herokuapp.com
3 X-Parse-Application-Id: 1ZQwgaTsfnduMHT0Q7HxAhVTekZocSpjz2isc
4 X-Parse-Session-Token: 1ZQwgaTsfnduMHT0Q7HxAhVTekZocSpjz2isc
5 X-Parse-App-Display-Version: 6.1.4
6 X-Parse-Os-Version: 12
7 User-Agent: Parse Android SDK API Level 31
8 X-Parse-Installation-Id: 0sd8ce5a-fael-4a75-9d6f-6ea6a67ff195
9 Content-Type: application/json
10 Content-Length: 39
11 Accept-Encoding: gzip, deflate
12 Connection: close
13
14 (
    "username": "ismail.arame66@gmail.com"
)

```

Inspector

Request Attributes: 2

Request Query Parameters: 0

Request Cookies: 0

Request Headers: 11

Sign Up

Email: ismail.arame66@gmail.com

Password: sayonara129

Length > 8 Uppercase Special Character Number

IMPORTANT: Your account password will be used to unlock mSecure.

ACCOUNT HELP? CONTINUE

- ✓ apres la confirmation du password on remarque que l'email et le password et intercepete par burpsuite en text clair, d'où un attacker qui est entrain d'espionner va avoir l'access à mon email et password

Burp Suite Professional v2022.11.4 - Temporary Project - licensed to surfenyz

Request to https://msecure-docker-parse.herokuapp.com:443 [23.22.130.173]

```

1 POST /parse/users HTTP/1.1
2 Host: msecure-docker-parse.herokuapp.com
3 X-Parse-Revocable-Session: 1
4 X-Parse-Application-Id: 1ZQwgaTsfnduMHT0Q7HxAhVTekZocSpjz2isc
5 X-Parse-Session-Token: 1ZQwgaTsfnduMHT0Q7HxAhVTekZocSpjz2isc
6 X-Parse-App-Display-Version: 6.1.4
7 X-Parse-Os-Version: 12
8 User-Agent: Parse Android SDK API Level 31
9 X-Parse-Installation-Id: 0sd8ce5a-fael-4a75-9d6f-6ea6a67ff195
10 Content-Type: application/json
11 Content-Length: 89
12 Accept-Encoding: gzip, deflate
13 Connection: close
14
15 (
    "password": "sayonara129",
    "email": "ismail.arame66@gmail.com",
    "username": "ismail.arame66@gmail.com"
)

```

Inspector

Request Attributes: 2

Request Query Parameters: 0

Request Cookies: 0

Request Headers: 12

Sign Up

Confirm Password: sayonara129

Password Hint: say129

ACCOUNT HELP? CREATE ACCOUNT

- ✓ D'après mSecure, lors de mon premier login, mSecure génère un super secret password appelé "Account Key" et il l'utilise pour chiffrer un text clair connu et ils le stocke sur mon compte

✓ dans leurs system. Après mon mot de passe personnel est utilisé par suite pour chiffre l'account key, et enfin ils m'envoient un email contenant l'account key chiffré qui est nécessaire pour m'authentifier en tant que propriétaire de mon compte:

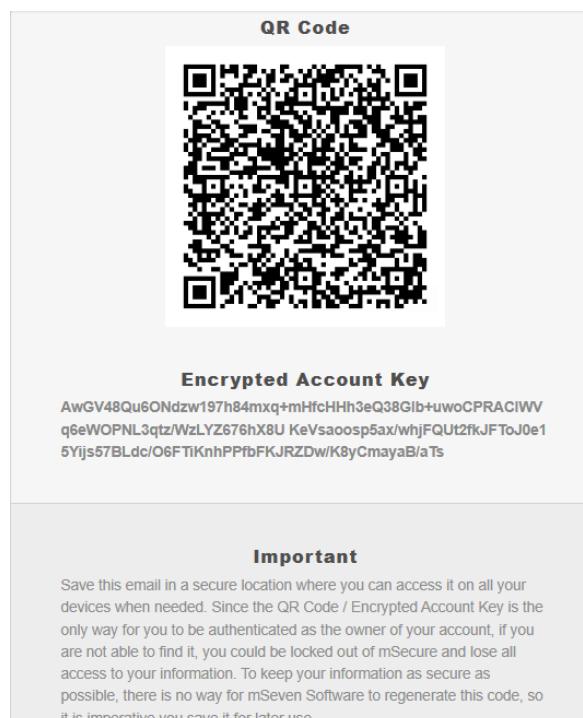
The screenshot shows the Burp Suite Professional interface with the 'Proxy' tab selected. A captured POST request is displayed in the 'Pretty' tab. The request body contains the following JSON data:

```

1 POST /parse/functions/generatePKey HTTP/1.1
2 Host: msecure-docker-parse.herokuapp.com
3 X-Parse-Session-Token: e-0e1321a35c-df02f4ab2e0892f98e31
4 X-Parse-Application-Id: 1ZCvqgCTF9ndmH7NTUQ7HxAhVTeK2ocSpjz5mc
5 X-Parse-App-Build-Version: 1872
6 X-Parse-App-Display-Version: 6.1.4
7 X-Parse-Os-Version: 12
8 User-Agent: Parse Android SDK API Level 31
9 X-Parse-Installation-ID: 05d9ce5a-fael-4a75-9d6f-6ea6a67ff19f
10 Content-Type: application/json
11 Content-Length: 220
12 Accept-Encoding: gzip, deflate
13 Connection: close
14
15 {
  "sendEmail":true,
  "passwordChange":false,
  "encryptedPKey":true
}
  
```

To the right, a screenshot of a mobile phone displaying a 'Sign In Successful' screen. The screen shows an email address (ismail.arame66@gmail.com) and a QR code. Below the QR code, there is an 'IMPORTANT' note stating: 'The QR code above is your encrypted account key used to authenticate you as the owner of your account. An mSecure Authentication email containing this code has been sent to the email address above. If you don't receive the email in your inbox, make sure to check your spam folder in case it was marked as junk mail.' A 'CONTINUE' button is at the bottom.

✓ l'email que j'ai recu.



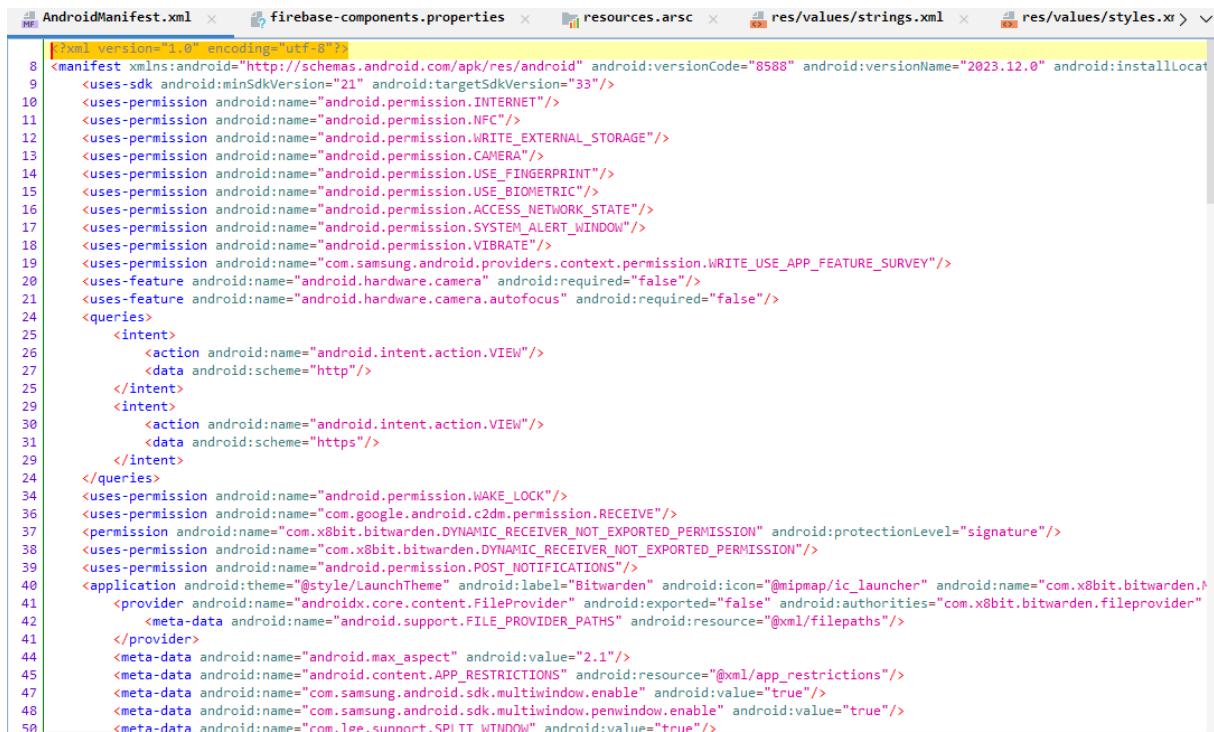
➤ Bitwarden :

⊕ Static analysis :

La première étape est de pull l'apk au disque externe de notre pc.

```
PS C:\Users\PC\Downloads\Hello>Hello> adb pull /data/app/~~U7lzYHquPH0l7TGQ9DE8Ww==/com.x8bit.bitwarden-Wr7d2DeUQ8RdK85Md8HCVw==/base.apk bitwarden.apk
/data/app/~~U7lzYHquPH0l7TGQ9DE8Ww==/com.x8bit.bitwarden-Wr7d2DeUQ8RdK85Md8HCVw==/base.apk: 1 file pulled, 0 skipped. 127.4 MB/s (26030332 bytes in 0.195s)
PS C:\Users\PC\Downloads\Hello>Hello>
```

Après la récupération du l'apk , on utilise jadx pour la décompilation et obtenir le code source et les fichiers de l'apk ,On commence par l'analyse du fichier Android.Manifest.XML



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android" android:versionCode="8588" android:versionName="2023.12.0" android:installLocat
8   <uses-sdk android:minSdkVersion="21" android:targetSdkVersion="33"/>
9   <uses-permission android:name="android.permission.INTERNET"/>
10  <uses-permission android:name="android.permission.NFC"/>
11  <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE"/>
12  <uses-permission android:name="android.permission.CAMERA"/>
13  <uses-permission android:name="android.permission.USE_FINGERPRINT"/>
14  <uses-permission android:name="android.permission.USE_BIOMETRIC"/>
15  <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
16  <uses-permission android:name="android.permission.SYSTEM_ALERT_WINDOW"/>
17  <uses-permission android:name="android.permission.VIBRATE"/>
18  <uses-permission android:name="com.samsung.android.providers.context.permission.WRITE_USE_APP_FEATURE_SURVEY"/>
19  <uses-feature android:name="android.hardware.camera" android:required="false"/>
20  <uses-feature android:name="android.hardware.camera.autofocus" android:required="false"/>
21  <queries>
22    <intent>
23      <action android:name="android.intent.action.VIEW"/>
24      <data android:scheme="http"/>
25    </intent>
26    <intent>
27      <action android:name="android.intent.action.VIEW"/>
28      <data android:scheme="https"/>
29    </intent>
30  </queries>
31  <uses-permission android:name="android.permission.WAKE_LOCK"/>
32  <uses-permission android:name="com.google.android.c2dm.permission.RECEIVE"/>
33  <permission android:name="com.x8bit.bitwarden.DYNAMIC_RECEIVER_NOT_EXPORTED_PERMISSION" android:protectionLevel="signature"/>
34  <uses-permission android:name="com.x8bit.bitwarden.DYNAMIC_RECEIVER_NOT_EXPORTED_PERMISSION"/>
35  <uses-permission android:name="android.permission.POST_NOTIFICATIONS"/>
36  <application android:theme="@style/LaunchTheme" android:label="Bitwarden" android:icon="@mipmap/ic_launcher" android:name="com.x8bit.bitwarden.>
37    <provider android:name="androidx.core.content.FileProvider" android:exported="false" android:authorities="com.x8bit.bitwarden.fileprovider">
38      <meta-data android:name="android.support.FILE_PROVIDER_PATHS" android:resource="@xml/filepaths"/>
39    </provider>
40    <meta-data android:name="android.max_aspect" android:value="2.1"/>
41    <meta-data android:name="android.content.APP_RESTRICTIONS" android:resource="@xml/app_restrictions"/>
42    <meta-data android:name="com.samsung.android.sdk.multiwindow.enable" android:value="true"/>
43    <meta-data android:name="com.samsung.android.sdk.multiwindow.penwindow.enable" android:value="true"/>
44    <meta-data android:name="com.lge.sunnsor_SPI_TT_WNDOW" android:value="true"/>
45  </application>
46</manifest>
```

```

62
63
64
65
66 />
67 id:exported="true" android:launchMode="singleTop" android:noHistory="true">
68
69
70
71
72
73
74
75 " android:name="com.x8bit.bitwarden.AutofillTileService" android:permission="android.permission.BIND_QUICK_SETTINGS_TILE" android:exported="true">
76
77 >
78
79
80 generate" android:name="com.x8bit.bitwarden.GeneratorTileService" android:permission="android.permission.BIND_QUICK_SETTINGS_TILE" android:exported="true">
81
82 >
83
84 >
85 >
86 >
87 >
88
89

```

✓ Res/values/strings.xml:

On trouve deux API_KEYs et un lien d'une database qui peut être usable pour découvrir une vulnérabilité.

The screenshot shows the Jadx-GUI interface with the file 'res/values/strings.xml' open. The left sidebar shows the project structure with 'values' and 'strings.xml' selected. The right pane displays the XML code for strings.xml. Two specific strings are highlighted with red boxes and arrows pointing to them:

- `<string name="gcm_defaultSenderId">64530857057</string>`
- `<string name="google_app_id">1:64530857057:android:f8d67b786db1b844</string>`

```

<resources>
    <string name="confirm_device_credential_password">Use password</string>
    <string name="copy">Copy</string>
    <string name="copy_toast_msg">Link copied to clipboard</string>
    <string name="default_error_msg">Unknown error</string>
    <string name="default_web_client_id">64530857057-pbul8wlpuvfrv7ju2dm2rjk0ah9tt1.apps.googleusercontent.com</string>
    <string name="error_a11y_label">Error: invalid</string>
    <string name="error_icon_content_description">Error</string>
    <string name="expand_button_title">Advanced</string>
    <string name="exposed_dropdown_menu_content_description">Show dropdown menu</string>
    <string name="fab_transformation_scrim_behavior">com.google.android.material.transformation.FabTransformationScrimBehavior</string>
    <string name="fab_transformation_sheet_behavior">com.google.android.material.transformation.FabTransformationSheetBehavior</string>
    <string name="fallback_menu_item_copy_link">Copy link</string>
    <string name="fallback_menu_item_open_in_browser">Open in browser</string>
    <string name="fallback_menu_item_share_link">Share link</string>
    <string name="fcm_fallback_notification_channel_label">Miscellaneous</string>
    <string name="fingerprint_dialog_touch_sensor">Touch the fingerprint sensor</string>
    <string name="fingerprint_error_hw_not_available">Fingerprint hardware not available.</string>
    <string name="fingerprint_error_hw_not_present">This device does not have a fingerprint sensor</string>
    <string name="fingerprint_error_lockout">Too many attempts. Please try again later.</string>
    <string name="fingerprint_error_no_fingerprints">No fingerprints enrolled.</string>
    <string name="fingerprint_error_user_canceled">Fingerprint operation canceled by user.</string>
    <string name="fingerprint_not_recognized">Not recognized</string>
    <string name="firebase_database_url">https://bitwarden-c2b35.firebaseio.com</string>
    <string name="gcm_defaultSenderId">64530857057</string>
    <string name="generic_error_no_device_credential">No PIN, pattern, or password set.</string>
    <string name="generic_error_no_keyguard">This device does not support PIN, pattern, or password.</string>
    <string name="generic_error_user_canceled">Authentication canceled by user.</string>
    <string name="google_api_key">AIzaSyCjmReBzwA23n5z1lUjg6ag6Nx-DNpKN_w</string>
    <string name="google_app_id">1:64530857057:android:f8d67b786db1b844</string>
    <string name="google_crash_reporting_api_key">AIzaSyCjmReBzwA23n5z1lUjg6ag6Nx-DNpKN_w</string>
    <string name="google_storage_bucket">bitwarden-c2b35.appspot.com</string>
    <string name="hide_bottom_view_on_scroll_behavior">com.google.android.material.behavior.HideBottomViewOnScrollBehavior</string>
    <string name="icon_content_description">Dialog Icon</string>
    <string name="item_view_role_description">Tab</string>
    <string name="library_name">ZXing.Net.Mobile.Forms.Android</string>
    <string name="m3_ref_typeface_brand_medium">sans-serif-medium</string>

```

✓ Analyse du code source :

```
/* Loaded from: classes.dex */
public static class Pre28Implementation implements SignaturesCompat {
    Pre28Implementation() {
    }

    @Override // androidx.browser.trusted.PackageIdentityUtils.SignaturesCompat
    public List<byte[]> getFingerprintsForPackage(String name, PackageManager pm) throws PackageManager.NameNotFoundException {
        PackageInfo packageInfo = pm.getPackageInfo(name, 64);
        ArrayList arrayList = new ArrayList(packageInfo.signatures.length);
        for (Signature signature : packageInfo.signatures) {
            byte[] certificateSHA256Fingerprint = PackageIdentityUtils.getCertificateSHA256Fingerprint(signature);
            if (certificateSHA256Fingerprint == null) {
                return null;
            }
            arrayList.add(certificateSHA256Fingerprint);
        }
        return arrayList;
    }

    @Override // androidx.browser.trusted.PackageIdentityUtils.SignaturesCompat
    public boolean packageMatchesToken(String name, PackageManager pm, TokenContents token) throws IOException, PackageManager.NameNotFoundException {
        List<byte[]> fingerprintsForPackage;
        if (name.equals(token.getPackageName()) && (fingerprintsForPackage = getFingerprintsForPackage(name, pm)) != null) {
            return token.equals(TokenContents.create(name, fingerprintsForPackage));
        }
        return false;
    }
}

static byte[] getCertificateSHA256Fingerprint(Signature signature) {
    try {
        return MessageDigest.getInstance("SHA256").digest(signature.toByteArray());
    } catch (NoSuchAlgorithmException unused) {
        return null;
    }
}
```

```
/* Loaded from: classes.dex */
class CryptoAesHandler implements CryptoHandler {
    @Override // com.microsoft.appcenter.utils.crypto.CryptoHandler
    public String getAlgorithm() {
        return "AES/CBC/PKCS7Padding/256";
    }

    @Override // com.microsoft.appcenter.utils.crypto.CryptoHandler
    public void generateKey(CryptoUtils.ICryptoFactory cryptoFactory, String alias, Context context) throws Exception {
        Calendar calendar = Calendar.getInstance();
        calendar.add(1, 1);
        CryptoUtils.IKeyGenerator keyGenerator = cryptoFactory.getKeyGenerator("AES", "AndroidKeyStore");
        keyGenerator.init(new KeyGenParameterSpec.Builder(alias, 3).setBlockModes("CBC").setEncryptionPaddings("PKCS7Padding").setKeySize(256));
        keyGenerator.generateKey();
    }

    @Override // com.microsoft.appcenter.utils.crypto.CryptoHandler
    public byte[] encrypt(CryptoUtils.ICryptoFactory cryptoFactory, int apiLevel, KeyStore.Entry keyStoreEntry, byte[] input) throws Exception {
        CryptoUtils.ICipher cipher = cryptoFactory.getCipher("AES/CBC/PKCS7Padding", "AndroidKeyStoreBCWorkaround");
        cipher.init(1, ((KeyStore.SecretKeyEntry) keyStoreEntry).getSecretKey());
        byte[] iv = cipher.getIV();
        byte[] doFinal = cipher.doFinal(input);
        byte[] bArr = new byte[iv.length + doFinal.length];
        System.arraycopy(iv, 0, bArr, 0, iv.length);
        System.arraycopy(doFinal, 0, bArr, iv.length, doFinal.length);
        return bArr;
    }
```

- ✓ Pour generer le cle de chiffrement Bitwarden utilise AES CBC mode et SHA256 Pour la signature de certificate.

✓ Keystore :

The screenshot shows the Android Studio code editor with four tabs at the top: OnBackPressedDispatcher, OnBackPressedDispatcherKt, OnBackPressedDispatcherOwner, and CryptoObjectUtils (the active tab). The code in the editor is as follows:

```
218     return new FingerprintManagerCompat.CryptoObject(mac);
219 }
220 if (Build.VERSION.SDK_INT >= 30 && cryptoObject.getIdentityCredential() != null) {
221     Log.e(TAG, "Identity credential is not supported by FingerprintManager.");
222 }
223 return null;
224 }

/* JADX INFO: Access modifiers changed from: package-private */
241 public static BiometricPrompt.CryptoObject createFakeCryptoObject() {
242     try {
243         KeyStore keyStore = KeyStore.getInstance(KEYSTORE_INSTANCE);
244         keyStore.load(null);
245         KeyGenParameterSpec.Builder createKeyGenParameterSpecBuilder = Api23Impl.createKeyGenParameterSpecBuilder(FAKE_KEY_NAME, 3);
246         Api23Impl.setBlockModeCBC(createKeyGenParameterSpecBuilder);
247         Api23Impl.setEncryptionPaddingPKCS7(createKeyGenParameterSpecBuilder);
248         KeyGenerator keyGenerator = KeyGenerator.getInstance("AES", KEYSTORE_INSTANCE);
249         Api23Impl.initKeyGenerator(keyGenerator, Api23Impl.buildKeyGenParameterSpec(createKeyGenParameterSpecBuilder));
250         keyGenerator.generateKey();
251         Cipher cipher = Cipher.getInstance("AES/CBC/PKCS7Padding");
252         cipher.init(1, (SecretKey) keyStore.getKey(FAKE_KEY_NAME, null));
253         return new BiometricPrompt.CryptoObject(cipher);
254     } catch (IOException | InvalidAlgorithmParameterException | InvalidKeyException | KeyStoreException | NoSuchAlgorithmException | NoSuchProviderException e) {
255         Log.w(TAG, "Failed to create fake crypto object.", e);
256         return null;
257     }
258 }
```

✓ URLs :

Dynamic analysis :

- ✓ Push frida server in our device virtuel

```
C:\Users\PC\Downloads>adb push "C:\Users\PC\Downloads\PassManagers\frida-server-16.1.10-android-x86" data/local/tmp  
[C:\Users\PC\Downloads\PassManagers\frida-server-16.1.10-android-x86] successfully pushed, 0 skipped. 81.2 MB/s (51684860 bytes in 0.607s)  
C:\Users\PC\Downloads>
```

- ✓ Turn frida server to an executable and exe it and connect to frida server.

```
130@generic_x86:/ $ cd data/local/tmp  
generic_x86:/data/local/tmp $ ls  
frida-server-16.1.10-android-x86  oat  
generic_x86:/data/local/tmp $ chmod 777 frida-server-16.1.10-android-x86  
generic_x86:/data/local/tmp $
```

```
27@generic_x86:/data/local/tmp # ls  
frida-server-16.1.10-android-x86  oat  
generic_x86:/data/local/tmp # ./frida-server-16.1.10-android-x86
```

```
::\Users\PC>frida-ps -U  
PID  Name  
---  
040  Chrome  
358  Gmail  
129  Google  
766  Google Play Store  
484  Messages  
554  Phone  
951  Photos  
011  YouTube  
388  adb  
394  android.hardware.audio@2.0-service  
403  android.hardware.biometrics.fingerprint@2.1-service  
395  android.hardware.camera.provider@2.4-service  
396  android.hardware.configstore@1.0-service  
397  android.hardware.drm@1.0-service  
398  android.hardware.drm@1.0-service.widevine  
399  android.hardware.gatekeeper@1.0-service  
031  android.hardware.gnss@1.0-service  
400  android.hardware.graphicsallocator@2.0-service  
401  android.hardware.graphics.composer@2.1-service  
337  android.hardware.keymaster@3.0-service  
032  android.hardware.media.omx@1.0-service  
402  android.hardware.power@1.0-service  
403  android.hardware.sensors@1.0-service  
404  android.hardware.wifi@1.0-service  
393  android.hidl.allocator@1.0-service  
188  android.process.acore  
895  android.process.media  
019  audioserver  
020  cameraserver  
764  com.android.chrome  
397  com.android.phone  
825  com.android.printspooler  
295  com.android.systemui  
250  com.android.vending
```

- ✓ Apres on va faire le même processus qu'on a déjà fait dans les applications précédent, créer un listener utilisant Burpsuite et configurer le proxy dans notre device and push the ca certificate.
- ✓ Apres fair tous ca on va bypass SSL pinning using frida.

```
C:\Users\Document\AppData\Roaming\Python\Python39\Scripts>frida --codeshare pcipolloni/universal-android-ssl-pinning-bypass-with-frida -U -f com.x8bit.bitwarden

[|_|] Frida 16.0.8 - A world-class dynamic instrumentation toolkit
> Commands:
[|_|]   help      -> Displays the help system
...   object?   -> Display information about 'object'
...   exit/quit -> Exit
...
...   More info at https://frida.re/docs/home/
...
...   Connected to Nexus 4 (id=192.168.72.101:5555)
pawned `com.x8bit.bitwarden`. Resuming main thread!
Nexus 4::com.x8bit.bitwarden ]->
.] Cert Pinning Bypass/Re-Pinning
+] Loading our CA...
o] Our CA Info: CN=PortSwigger CA, OU=PortSwigger CA, O=PortSwigger, L=PortSwigger, ST=PortSwigger, C=PortSwigger
+] Creating a KeyStore for our CA...
+] Creating a TrustManager that trusts the CA in our KeyStore...
+] Our TrustManager is ready...
+] Hijacking SSLContext methods now...
..] Waiting for the app to invoke SSLContext.init()
```

- ✓ On a réussi de faire bypass SSL pinning et voila la resultat ,on remarque le traffic .

Burp Suite Community Edition v2023.12.1-25776 (Early Adopter) - Temporary Project

Proxy

Request to https://firebaseinstallations.googleapis.com:443 [172.217.168.170]

Forward Drop Intercept is on Action Open browser

Pretty Raw Hex

```
1 POST /v1/projects/bitwarden-cCb35/installations HTTP/1.1
2 Content-Type: application/json
3 Accept: application/json
4 Content-Encoding: gzip
5 Cache-Control: no-cache
6 X-Android-Package: com.x8bit.bitwarden
7 x-firebase-client: HAAAAAAAAMCMyhhNLcpJShOsKVayio7VUUSpLLSr0zMSTs1IyUqoFAPyivEQfAAAA
8 X-Android-Cert: 754105CD4CDPDE599748B043046BFES5A17264C2
9 x-goog-api-key: AIzaSyCjRkEBzwA23nfslluJg6ag6HX-DNpKRH_w
10 User-Agent: Dalvik/2.1.0 (Linux; U; Android 8.0.0; Android SDK built for x86 Build/OSR1.170901.043)
11 Host: firebaseinstallations.googleapis.com
12 Connection: close
13 Accept-Encoding: gzip, deflate, br
14 Content-Length: 133
15
16 D@WJELQ+RJ.UL+i-wog0i(+oi-rCIR0QJ,
17 (8)0'23156*05705.JIK)EIL&J*H1302.0KICL=0li(-EK-*iiIjs6D/3
18 Sd#D-I680DjzDpD
```

Event log

Filter Critical Error Info Debug

Time	Type	Source	Message
22:40:38 4 Jan 2024	Error	Proxy	[76] The client failed to negotiate a T
22:40:13 4 Jan 2024	Error	Proxy	[16] No response received from remo

Burp Suite Community Edition v2023.12.1-25776 (Early Adopter) - Temporary Project

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Intercept HTTP history WebSockets history | Proxy settings

Request to https://in.appcenter.ms:443 [52.232.209.85]

Forward Drop Intercept is on Action Open browser

Pretty Raw Hex

```

1 POST /logs?api-version=1.0.0 HTTP/1.1
2 Install-ID: 94a9e2a8-6091-478e-9640-7e6bbec851d5
3 App-Secret: d3834185-b4a6-4347-9047-b86c65293d42
4 Content-Type: application/json
5 Content-Length: 1130
6 User-Agent: Dalvik/2.1.0 (Linux; U; Android 8.0.0; Android SDK built for x86 Build/OSR1.170901.043)
7 Host: in.appcenter.ms
8 Connection: close
9 Accept-Encoding: gzip, deflate, br
10
11 {"logs": [{"type": "handledError", "timestamp": "2024-01-08T01:50:44.098Z", "device": {"wrapperSdkVersion": "5.0.2", "wrapperSdkName": "appcenter.xamarin", "wrapperRuntimeVersion": "13.2.2.0", "sdhName": "appcenter.android", "sdhVersion": "5.0.1", "model": "Android SDK built for x86", "osName": "Google", "osName": "Android", "osVersion": "8.0.0", "osBuild": "OSR1.170901.043", "osApiLevel": 26, "locale": "en_US", "timeZoneOffset": 0, "screenSize": "1080x1754", "appVersion": "2023.12.0", "carrierName": "Android", "carrierCountry": "us", "appBuild": "8580", "appNamespace": "com.x8bit.bitwarden"}, "id": "1460a511-9080-4be1-a003-92cbf2d80afe", "exception": {"type": "System.InvalidOperationException", "message": "Nullable object must have a value.", "stackTrace": " at System.Nullable`1[T].get_Value () [0x00008] in <8ed2c985ab6248558b7lc49fe7a4d008>:0 \n at Bit.Core.Services.EnvironmentService.SetUrlsFromStorageAsync () [0x001e1] in <02f6a334c19d4ff5b4f0758aae6accf16>:0 \n at Bit.App.Utilities.AccountManagement.AccountsManager.OnMessage (Bit.Core.Models.Domain.Message message) [0x000c0] in <1fd800d5c434beba8b832a467765093>:0 ", "wrapperSdkName": "appcenter.xamarin"}]}}

```

✓ voila une requête http n'est pas chiffré ,on peut utiliser ça pour être MITM.

Burp Suite Community Edition v2023.12.1-25776 (Early Adopter) - Temporary Project

Dashboard Target Proxy Intruder Repeater Collaborator Sequencer Decoder Comparer Logger Organizer Extensions Learn

Intercept HTTP history WebSockets history | Proxy settings

Request to https://android.apis.google.com:443 [142.250.200.110]

Forward Drop Intercept is on Action Open browser

Pretty Raw Hex

```

1 POST /cdm/register3 HTTP/2
2 Host: android.apis.google.com
3 Authorization: AidLogin 3834834573623759666:6897047595736832672
4 App: com.x8bit.bitwarden
5 Gcm-Ver: 11743470
6 Gcm-Cert: 2254a65b43e35fd14ba057bf65811a8aldda46f1
7 User-Agent: Android-GCM/1.5 (generic_x86 OSR1.170901.043)
8 Content-Length: 421
9 Content-Type: application/x-www-form-urlencoded
10 Connection: Keep-Alive
11 Accept-Encoding: gzip, deflate, br
12
13 X-subtype=645308570574X-app_ver=8588X-kid=7CID7C17C4X-osv=26X-cliv=fcm-23.1.2X-gmsv=11743470X-appid=eM_X5_xOR5exIaxWwDJcpF&X-scope=*X-gmp_app_id=13A6453085705743Android43Af8d67b706db1b844X.firebaseio-app-name=hash=RldAH9Ui7M-ynoznwBdw0ltLxh1&X-app_ver_name=2023.12.0X-app=com.x8bit.bitwarden&sender=645308570574device=3834834573623759666&cert=2254a65b43e35fd14ba057bf65811a8aldda46f1&app_ver=8588&gcm_ver=11743470

```