# **Les exploits pour l'acquisition**

**Module :** Analyse d'évidence pour terminaux mobiles



**Présentée Par :** **Encadré par :**

Alaoui Belghiti Hanaa

koulali mohammed amine

Slimani Hnia

Bouhsaine Laila

Sommaire

Contents

[**Les exploits pour l'acquisition** 1](#_Toc155646803)

[Exploit numéro 1 : Épuisement des ressources 3](#_Toc155646804)

[1) Connection d'appareil au Pc : 3](#_Toc155646805)

[2) Créer une scripte d'épuisement de ressources : 4](#_Toc155646806)

[3) Transférer cette script à l'appareil mobile : 4](#_Toc155646807)

[4) Exécution du script pour entrainer l'épuisement de ressources : 5](#_Toc155646808)

[5) Vérification des privilèges : 5](#_Toc155646809)

[6) Essayer des acquisitions : 5](#_Toc155646810)

[Exploit numéro 2 : Obtenir l'accès a distance au télé Android avec metasploit. 7](#_Toc155646811)

[1) Installation de Ngrok : 7](#_Toc155646812)

[2) Générer un Payload : 8](#_Toc155646813)

[3) Créer un écouteur dans Metasploit : 8](#_Toc155646814)

[4) Lancez l'attaque : 9](#_Toc155646815)

[5) Ouverture d'une session : 10](#_Toc155646816)

[6) Acquisition des données de l'appareil : 11](#_Toc155646817)

[1) Liste des applications : 11](#_Toc155646818)

[2) Envoyer des sms apartir de la session : 12](#_Toc155646819)

[3) Lister les contacts de l'appareil : 13](#_Toc155646820)

[4) Prise des photos : 14](#_Toc155646821)

[Exploit 3 : Phonesploit pro : 15](#_Toc155646822)

[1. C'est quoi PhoneSploit Pro 15](#_Toc155646823)

[2. Configuration du téléphone Android 16](#_Toc155646824)

[3. Nous allons essayer les commandes les plus importantes pour l’acquisition : 16](#_Toc155646825)

[4. Acceder au shell de l'appareil : 20](#_Toc155646826)

[5. Hacker l'appareil : 21](#_Toc155646827)

[6. Vulnérabilité Android dans ES File Explorer CVE-2019-6447 21](#_Toc155646828)

[7. Définitions 21](#_Toc155646829)

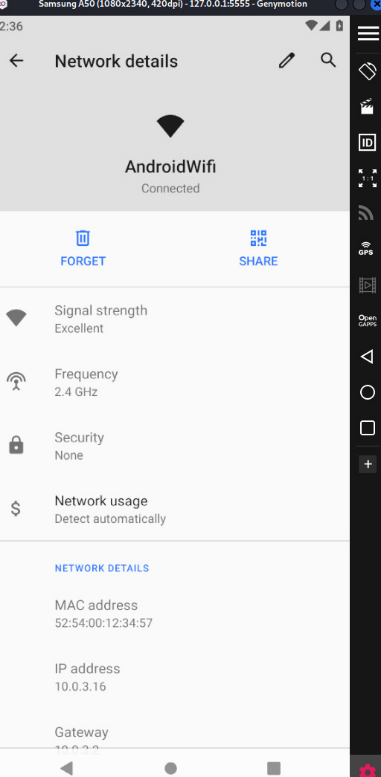
[8. Configuration : 22](#_Toc155646830)

[9. Exploitation : 22](#_Toc155646831)

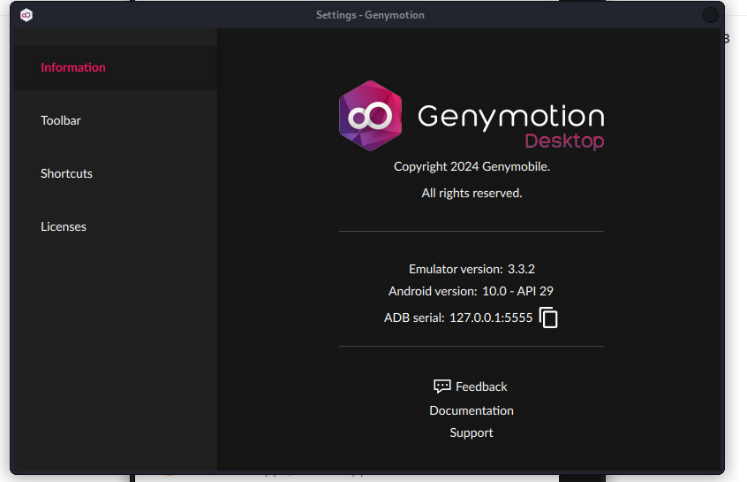
# Exploit numéro 1 : Épuisement des ressources

Cette attaque vise à générer un grand nombre de processus en arrière-plan sur un appareil Android dans le but d'obtenir un accès root non autorisé.

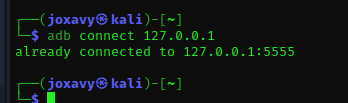
## Connection d'appareil au Pc :



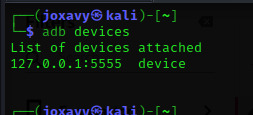
La version d'android utlisé est : 10



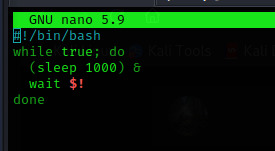
Connecter l'appareil :



Qui donne :



## Créer une scripte d'épuisement de ressources :

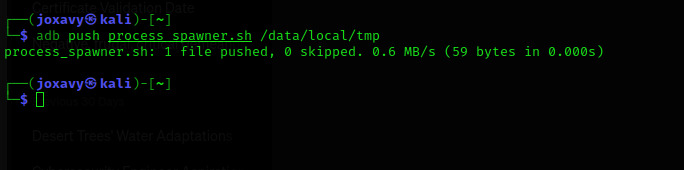


Nommée process\_spawner.sh

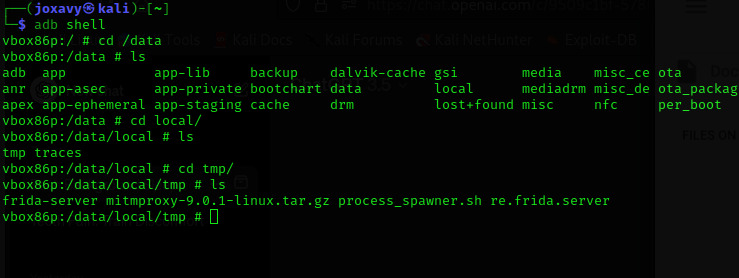
Que fait cette script ?

C'est une script Bash conçu pour générer un grand nombre de processus en arrière-plan sur un appareil Android. Le script crée une boucle infinie qui lance de nouveaux processus en arrière-plan, chacun avec une durée de sommeil de 1000 secondes. Cela peut potentiellement entraîner un épuisement des ressources système et une instabilité, car il consomme des ressources en créant un nombre excessif de processus.

## Transférer cette script à l'appareil mobile :

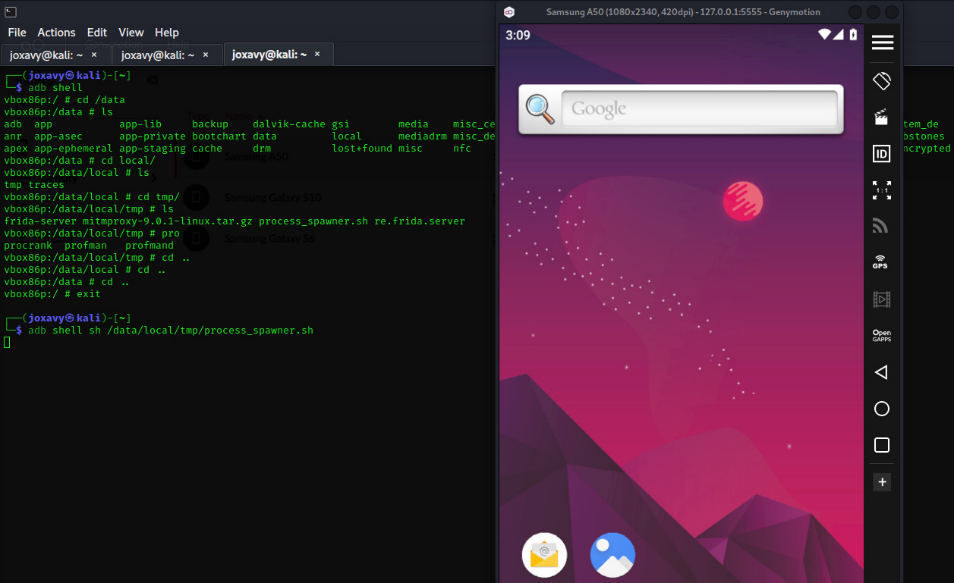


Vérifier si il existe dans l'appareil:



Script est maintenant dans l'appareil.

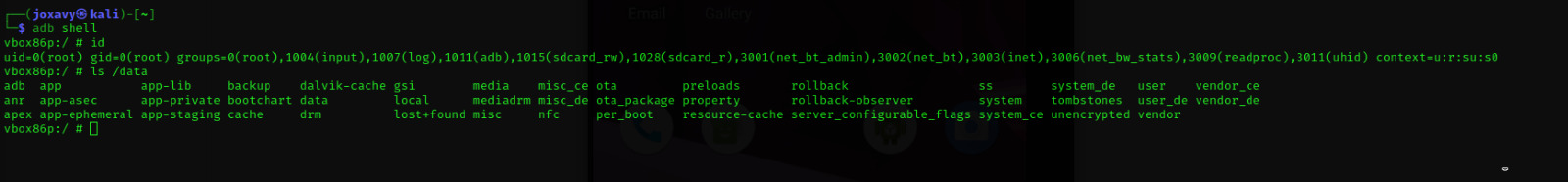
## Exécution du script pour entrainer l'épuisement de ressources :



Maintenant on doit attendre quelques moments avant de vérifier si on a l'Access root.

## Vérification des privilèges :

Lancer la commande suivante :



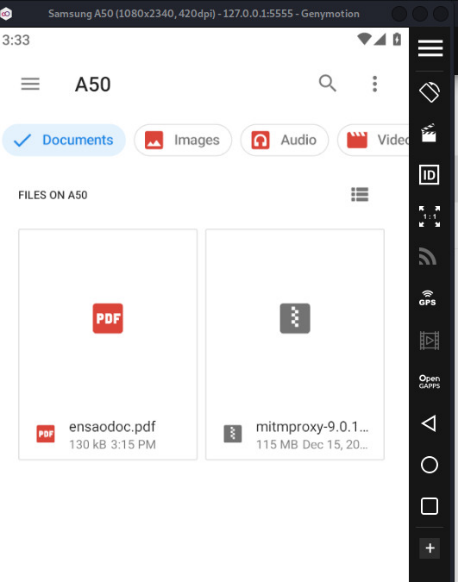
Résultat de cette commande donne l'id 0, qui est root.

et comme vous pouvez le voir, j'ai l'accès à les fichiers de la répertoire /data.

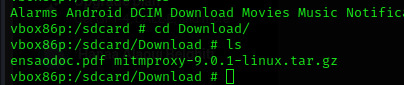
Et voilà !

## Essayer des acquisitions :

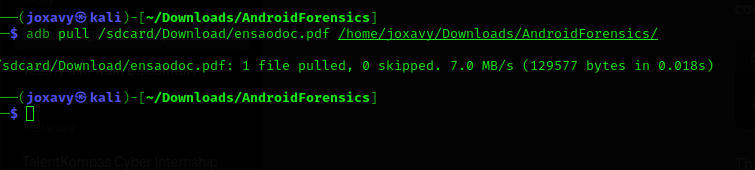
Premièrement je vais installer un document dans la télé.



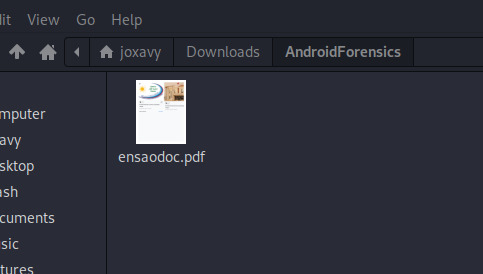
Qui est nommé ensaodoc.



Je vais utiliser la commande suivante pour faire un pull :



Et maintenant, le fichier est dans mon pc !



L'acquisition et un succès !

# Exploit numéro 2 : Obtenir l'accès a distance au télé Android avec metasploit.

Cette attaque est utilisée pour obtenir un accès à distance et créer une session sur un appareil mobile Android en utilisant le Framework Metasploit.

## **Installation de Ngrok :**

C'est quoi Ngrok ?

Ngrok est un proxy inverse qui crée un tunnel sécurisé depuis un point de terminaison public vers un service Web exécuté localement.

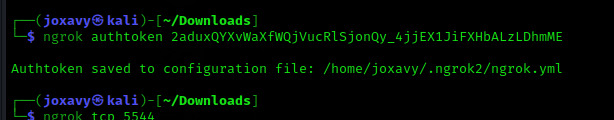
Il est utilisé pour créer un tunnel sécurisé et établir une connexion entre l'appareil compromis et un serveur distant contrôlé.

Ce tunnel permet à l'attaquant d'exposer à Internet les services locaux exécutés sur l'appareil compromis, facilitant ainsi l'accès et le contrôle à distance.

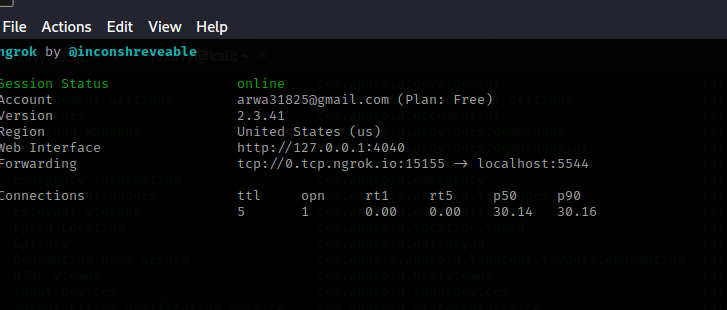
Installer et unzipper :



Authentifiez l'agent ngrok. Le jeton d'authentification est enregistré dans le fichier de configuration par défaut.



Maintenant, la connexion est établie, Ngrok a créé un transitaire de port : :

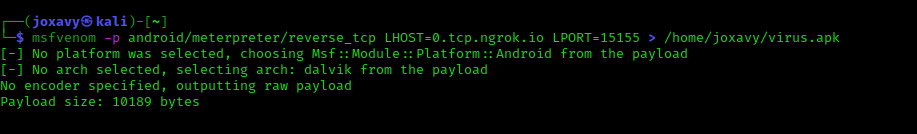


Mon adresse IP est : 0.tcp.ngrok.io

Cela signifie que si nous essayons d'accéder à l'URL 0.tcp.ngrok.io:15155, Ngrok affichera la réponse de mon hôte local et le port localhost est 5544.

## Générer un Payload :

Pour créer une "application" malveillante



Il est essentiellement un morceau de code qui, lorsqu'il est exécuté sur un appareil Android, établit une connexion TCP inversée vers ma machine.

Ici :

-p indique un type de charge utile

android/metepreter/reverse\_tcp spécifie qu'un shell de compteur inversé viendrait d'un appareil Android cible

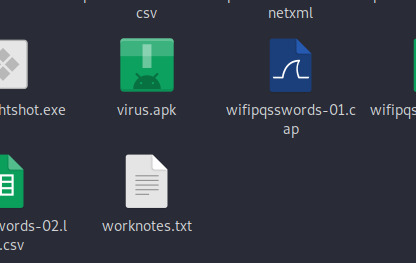
LHOST est l'IP locale

LPORT est le port d'écoute de l'IP

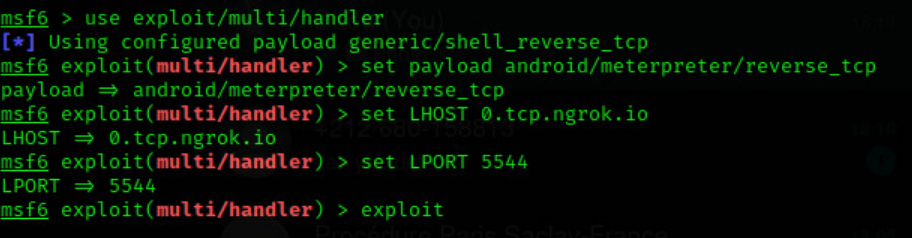
/home/user/ donnerait la sortie directement

apk est la dernière application malveillante.

Voilà le résultat obtenu dans le pc :



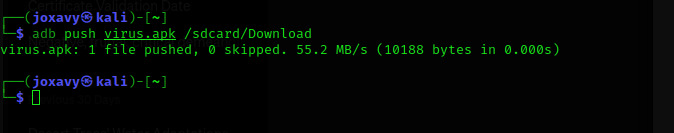
## Créer un écouteur dans Metasploit :



En attend de connexion.

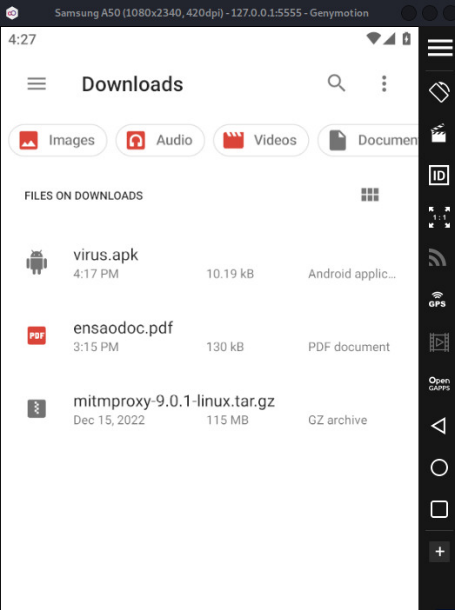
## Lancez l'attaque :

Installons maintenant le « virus.apk » sur l’appareil ciblé et ouvrons l’application.

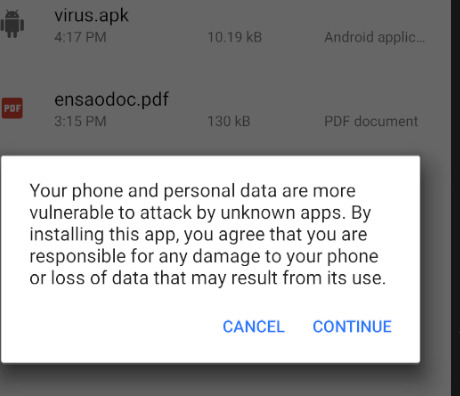
Quand l'utilisateur ouvre l'application, il enverra une connexion à notre Listener et créera une session. 

L'application est maintenant dans l'appareil.

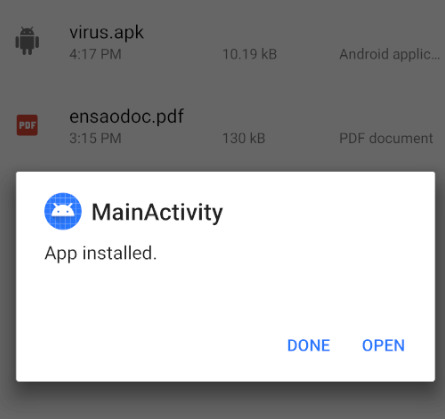
Voila :



Lancer l'application :

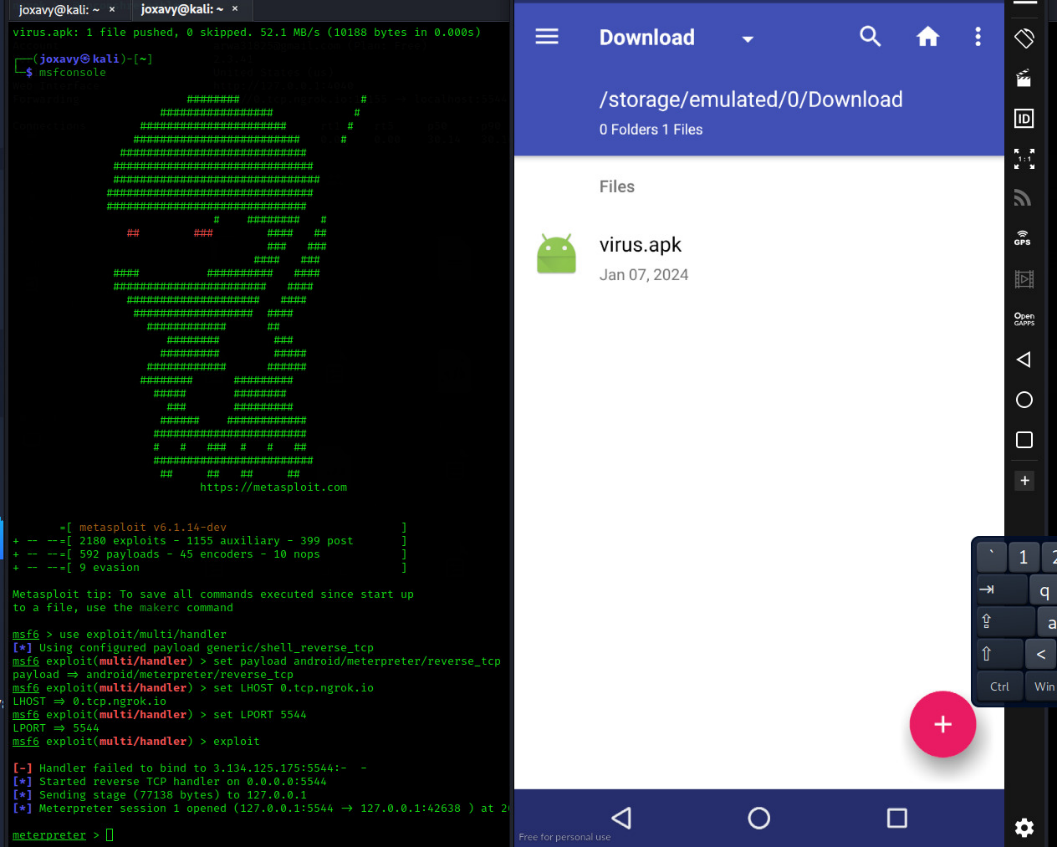


Est



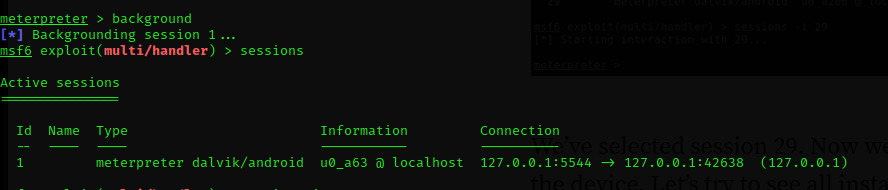
## Ouverture d'une session :

Apres l'installation de l'application et l'ouverture on obtient :



Une session a été créerer!

C'est la suivante :



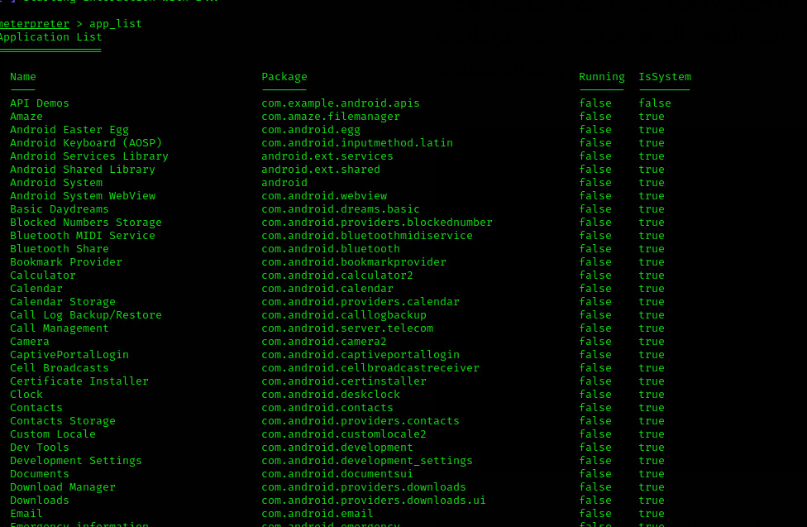
Avec notre appareil mobile.

Ouvrir la session :

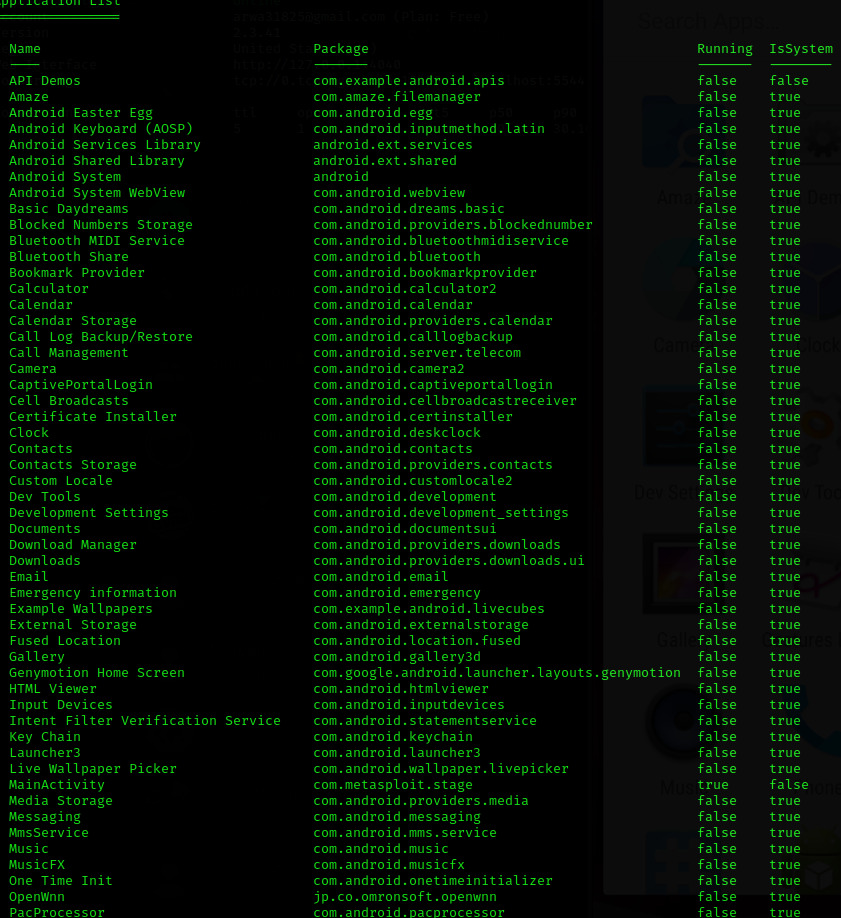


## Acquisition des données de l'appareil :

### Liste des applications :

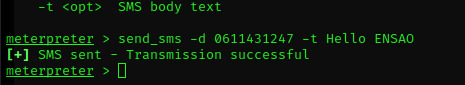


Cette commande nous donne la liste des appareils installer dans l'appareil.

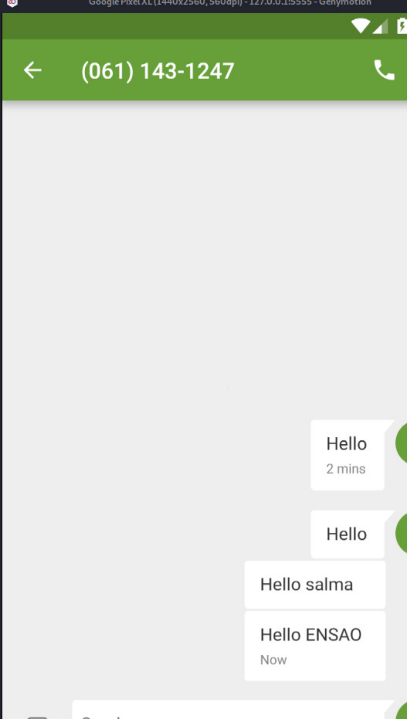


Mainactivity est notre application mallveinete.

### Envoyer des sms apartir de la session :



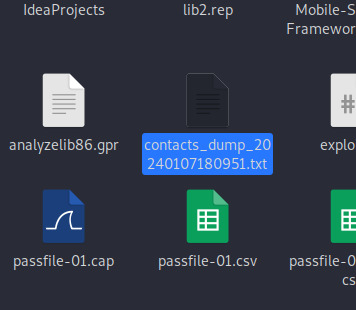
Qui donne :



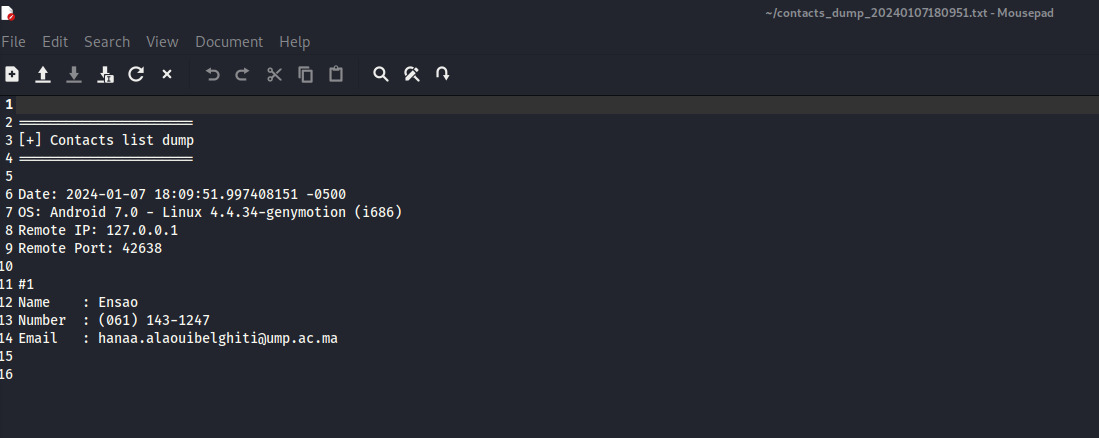
### Lister les contacts de l'appareil :



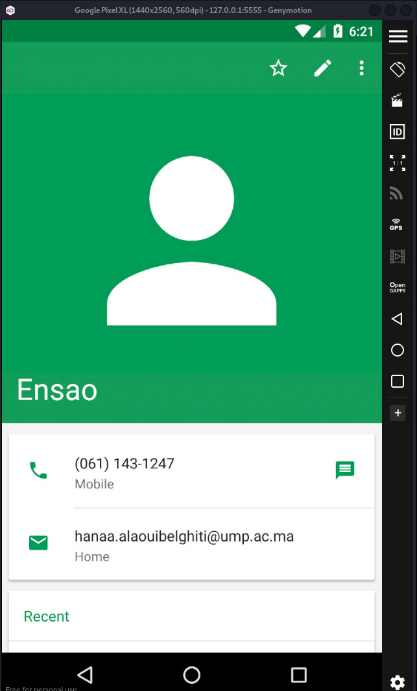
Qui donne le résultat suivant en ouvrant ce fichier :



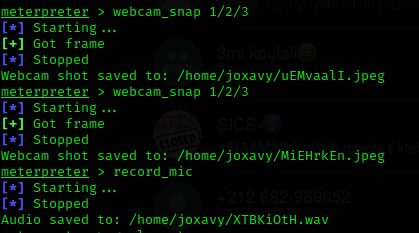
En ouvrant ce fichier :



Qui est le seul contact de l'appareil :



### Prise des photos :



Résultat est enregistrer dans mon dossier.

On peut faire tous ce qu'on fait.

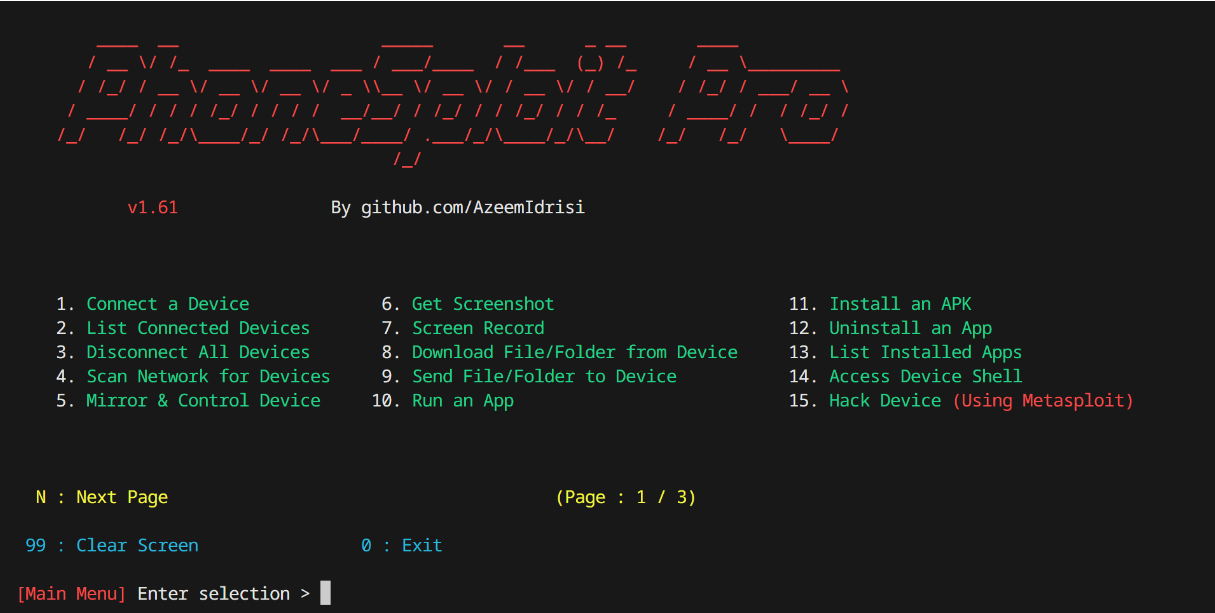
Donc l'acquisition et l'exploit sont un succès.

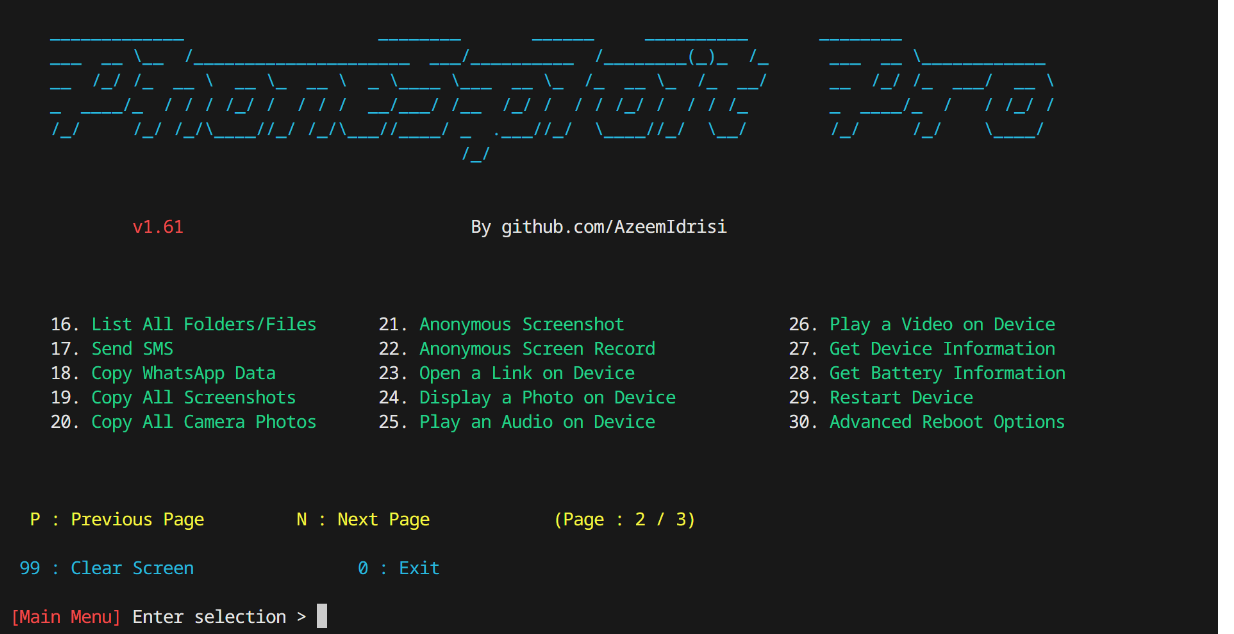
# Exploit 3 : CVE-2019-6447

## C'est quoi PhoneSploit Pro

PhoneSploit Pro est un puissant outil open source écrit en langage Python pour exploiter à distance des appareils Android en utilisant ADB (Android Debug Bridge) et le framework Metasploit.

Cet outil peut pirater complètement un smartphone Android en créant automatiquement, installant et exécutant une charge utile sur le dispositif cible. Ce projet vise à simplifier les tests de pénétration des appareils Android.







## Configuration du téléphone Android

Nous allons activer les options pour les développeurs.

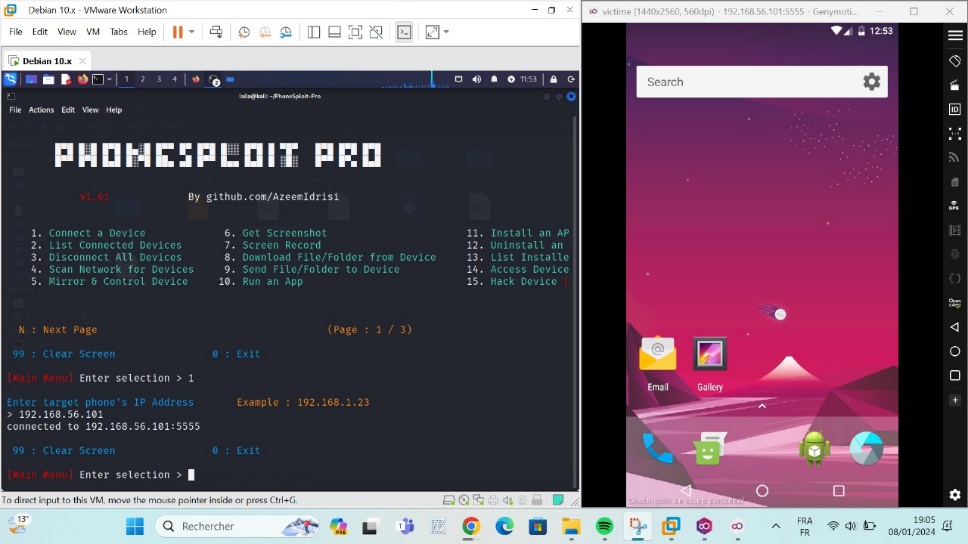
Puis on active le débogage USB, et on connecter l’appareil Android et l’ordinateur hôte adb au même réseau Wi-Fi.

Vérification :

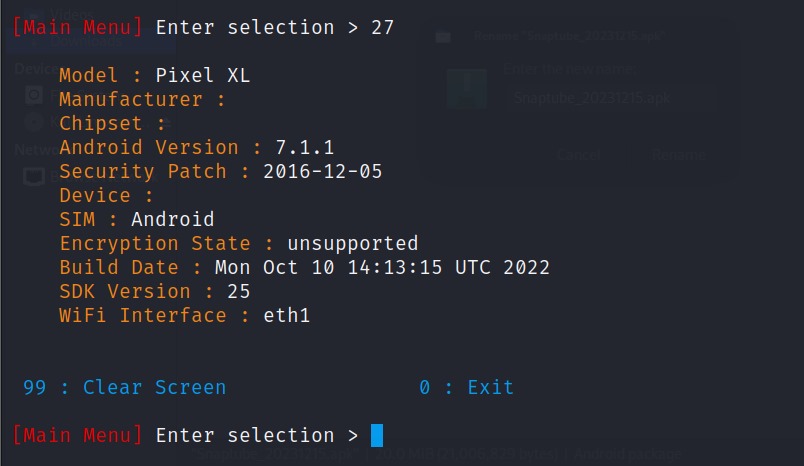


## Nous allons essayer les commandes les plus importantes pour l’acquisition :

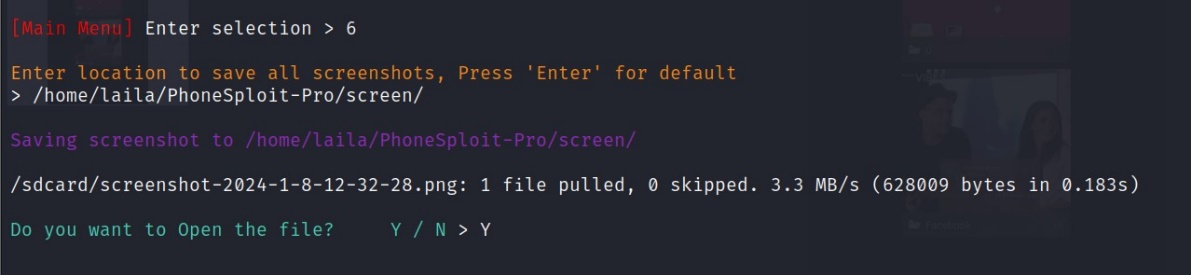
* Connecter l'appreil:

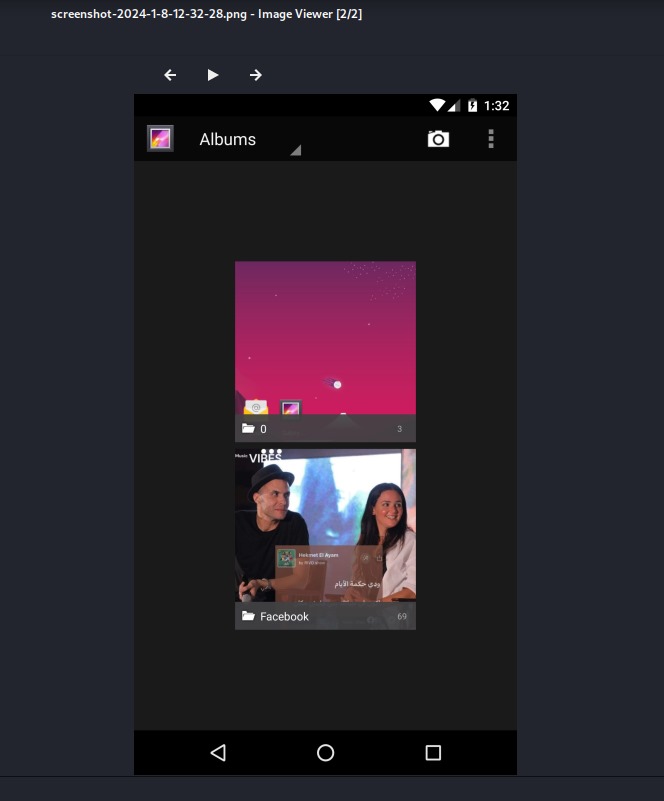


* Les infos de l'appareil :

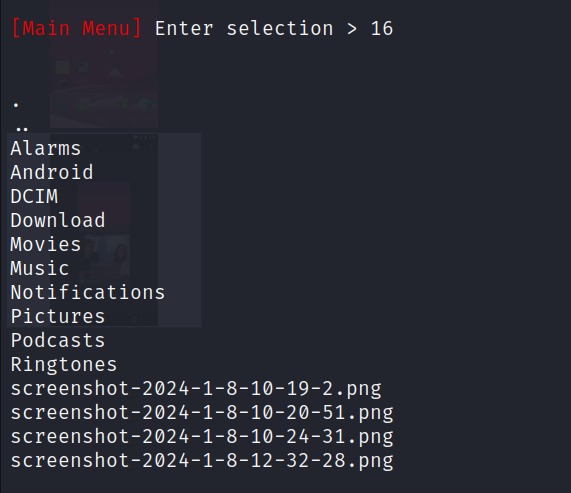


* Prendre une screenshot depuis le tele:

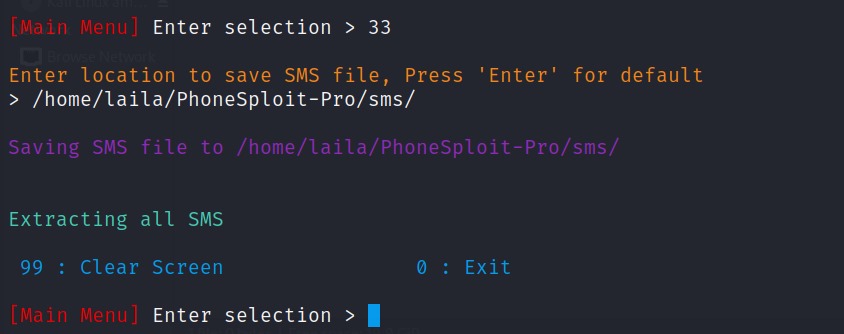


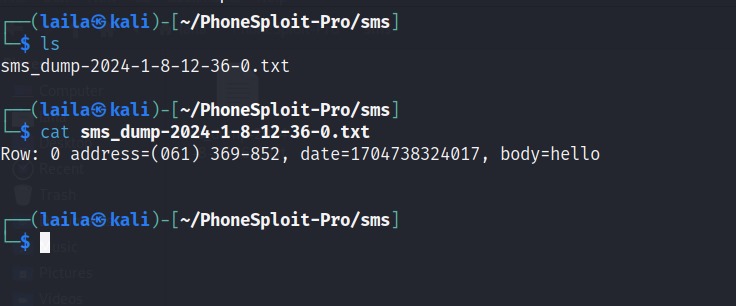


* Lister tous les dossiers :

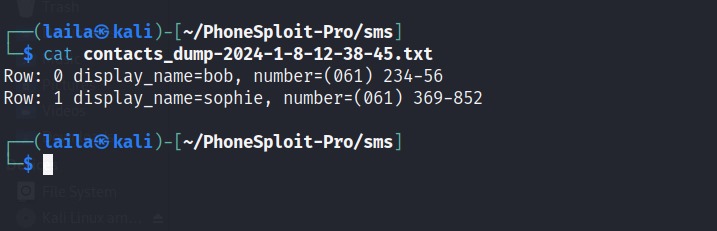


* Dump les sms :

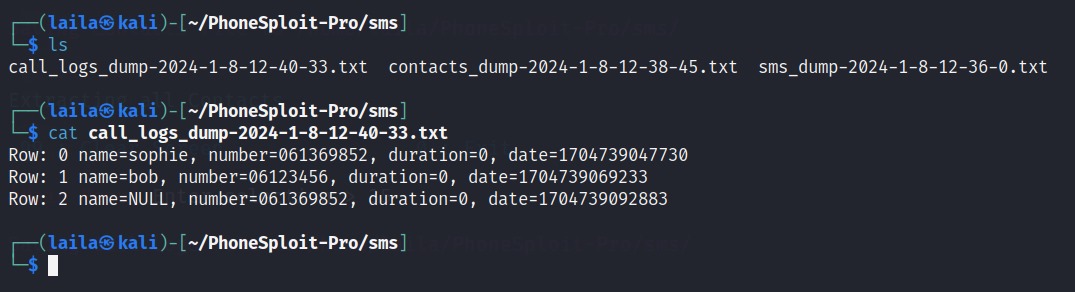




* Dump les conctacts :

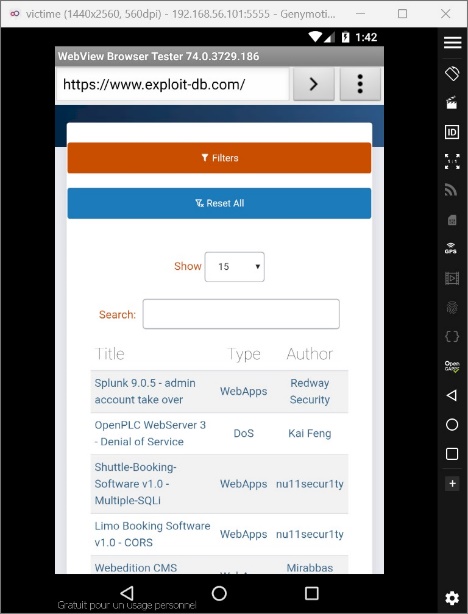


* Dump les call logs :

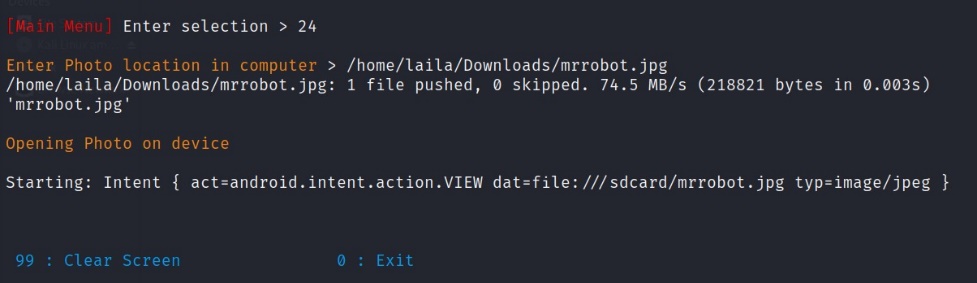


* Ouvrir un lien dans l'appareil depuis mon shell :



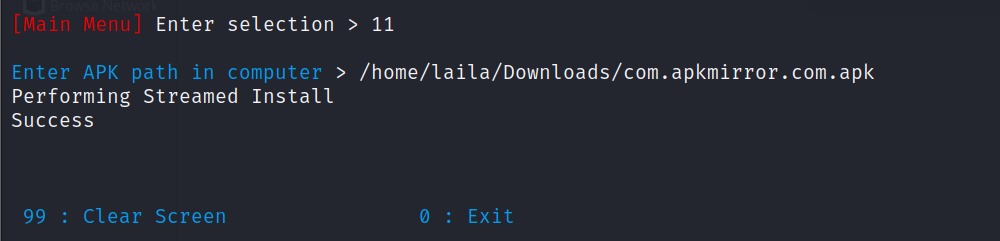


* Afficher une photo dans l'appareil:

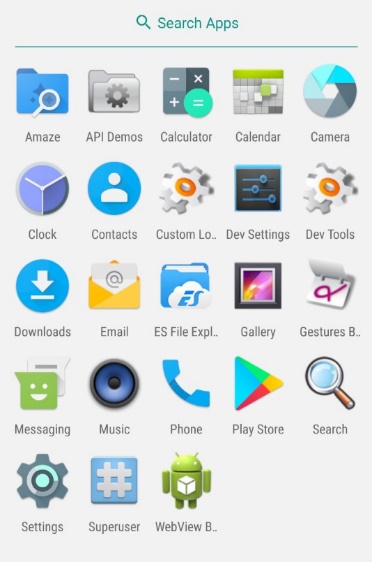




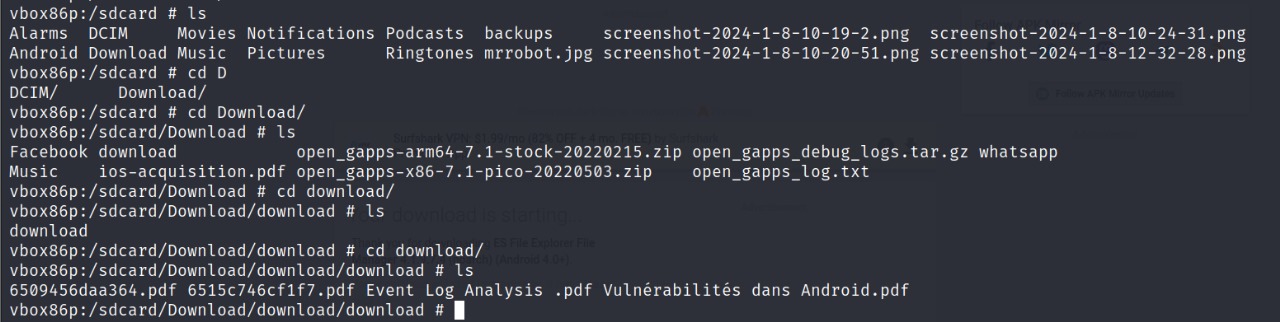
* Installer un apk :



Résultat:



## Acceder au shell de l'appareil :



## Hacker l'appareil :







## Vulnérabilité Android dans ES File Explorer CVE-2019-6447

Cette partie mettra en lumière l'exploitation manuelle de la vulnérabilité d'ES File Explorer.

Cela fonctionne sur la version v4.1.9.7.4. Il permet aux attaquants sur le même réseau d'exécuter des applications, de lire des fichiers et des données personnelles sensibles.

L'application laisse le port TCP 59777 ouvert pendant l'exécution et répond aux demandes contrefaites via http. Nous réaliserons cela dans un environnement virtuel avec une démonstration pratique pour une meilleure compréhension.

## Définitions

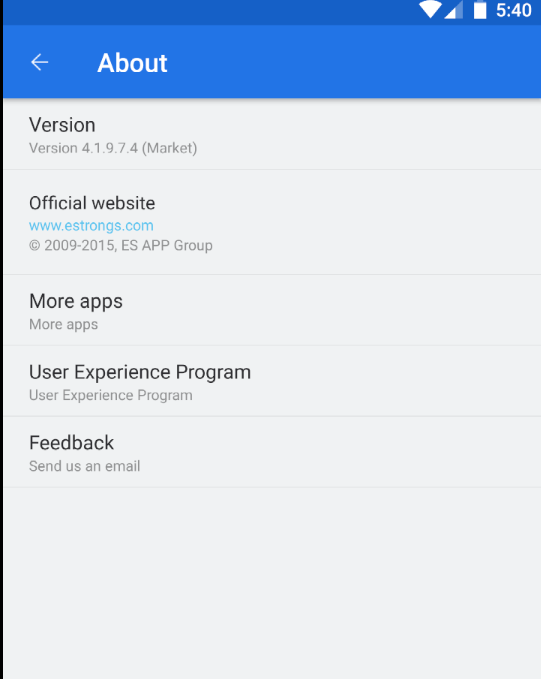
**ES File Explorer** Un gestionnaire de fichiers développé par une filiale de DO Global, c'est-à-dire ES Global. C'est le gestionnaire de fichiers le plus populaire sur Android avec plus de 100 millions d'installations. Il a été retiré du Play Store en raison de fraudes au clic.

**Metasploit Framework** Il est détenu par Rapid7, une entreprise de sécurité basée à Boston, Massachusetts. C'est un framework open source basé sur Ruby, utilisé comme aide aux tests de pénétration par les professionnels de la sécurité DevSecOps et les hackers éthiques.

## Configuration :

Ma configuration est la suivante :

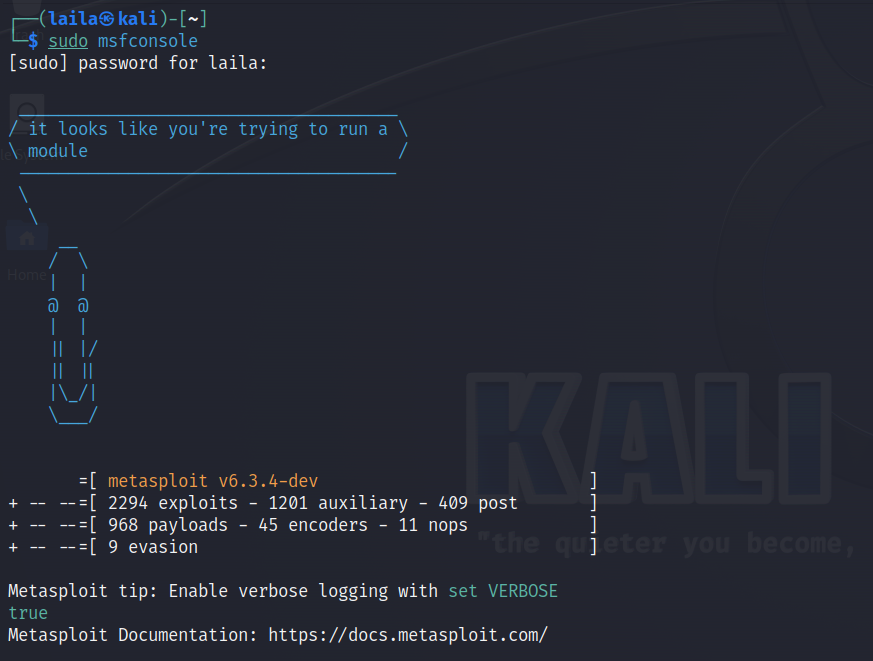
* Kali Linux sur virtualbox en tant que machine d'attaque.
* J'ai téléchargé la version d'ES Explorer affectée, la version 4.1.9.7.4, et l'ai installée sur mon téléphone. (la machine victime)



* Maintenant, les deux machines doivent être sur le même réseau.
* De plus, l'application doit être en cours d'exécution sur le téléphone afin que le port 59777 reste ouvert.

## Exploitation :

J'ai exécuté la commande **msfconsole** dans le terminal de Kali en tant que superutilisateur.

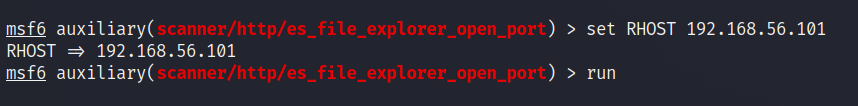


J'ai effectué une recherche d'exploit pour ES File Explorer pour vérifier si un exploit était disponible dans la base de données de Metasploit. Heureusement, il était disponible !



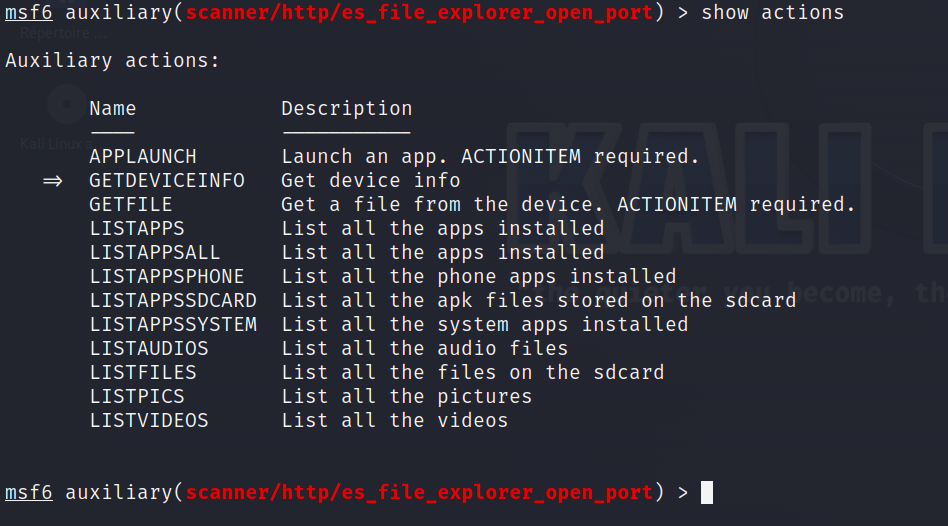
Comme nous pouvons le voir, RPORT est déjà réglé sur 59777. Cependant, nous devons définir RHOST.

Maintenant, nous devons exécuter la commande **set RHOST <ADRESSE\_IP>**, qui dans ce cas est l'adresse IP de mon téléphone.

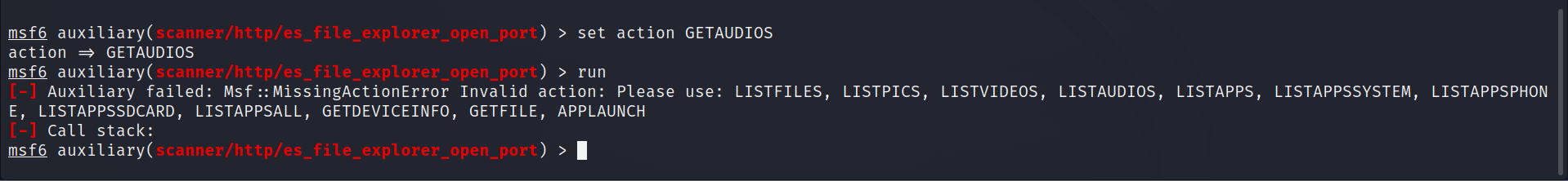


Ensuite, j'ai exécuté la commande "run" et j'ai pu voir les détails de mon téléphone mobile.

Maintenant, nous allons exécuter la commande "show actions" pour vérifier les actions disponibles que nous pouvons effectuer sur le périphérique.

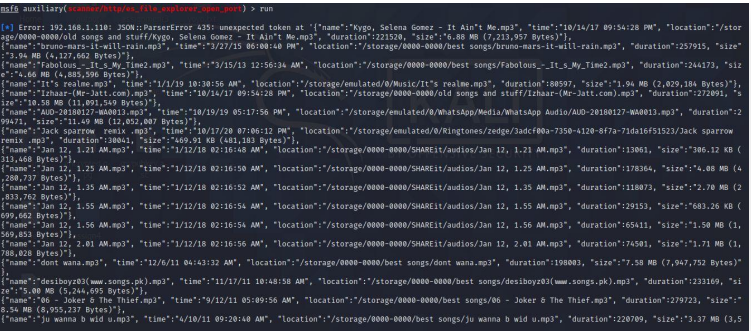


Maintenant, parmi celles-ci, nous voulons accéder aux fichiers audio. Donc, exécutez la commande **set action LISTAUDIOS**

****

Comme vous pouvez le remarquer, La partie où nous devions obtenir des fichiers audio n'a pas produit les résultats escomptés.

Cependant, je vais vous montrer un exemple réussi provenant d'une autre source pour illustrer le résultat correct.

****

Comme vous pouvez le voir dans la capture d'écran ci-dessus, nous avons obtenu un fichier audio personnel d'un utilisateur sans la permission ni la connaissance de l'utilisateur.

Maintenant, il ne s'agissait que d'un fichier audio ; nous pouvons accéder à n'importe quoi, comme des vidéos, des fichiers, et en plus, nous pouvons lancer une application à distance.

**Mesures d'atténuation**

* ES Explorer a publié de nombreuses versions corrigées après que le bug/vulnérabilité ait été signalé à la fin de 2019.
* Il est conseillé aux utilisateurs de mettre à jour vers la dernière version.
* Actuellement, la version 4.2.4.6.3 est la plus récente.

**FIN**