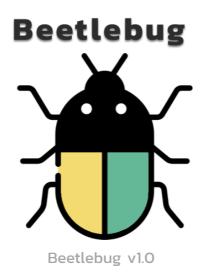
# A very insecure Android CTF App

# Lab Setup Guide



Author: Hafiz Adaji-Aziz

Email: contact@hafiz.ng March, 2022

To get started with Beetle CTF challenges, you will need to set up your Lab environment. These tools are requirements for conducting Android penetration testing.

# **Tools Required**

- Genymotion
- ADB (Android Debug Bridge)
- MobSF
- JADX Java Decompiler
- Drozer

# **Android Debug Bridge**

ADB is command line utility that facilitates communication between your Android device and a PC.

## How to set up ADB

To use ADB on your system, you first need to download and unzip the Android SDK Platform Tools from the Android developer website.

Mac users can use Homebrew to install ADB using the command: brew install/homebrew/cask/android-platform-tools

On your Android device, you will need to enable USB debugging in the developer settings before you can connect the device over ADB.

Go to Setting > About phone then tap on the Build number several times and wait to the message "You are now a developer!". Once again, go to Settings > Developer options, then toggle on USB debugging.

Now connect the Android device to your PC using USB cable. Always ensure to allow the USB debugging prompt that will appear on your Android device. Open the command line or Terminal and navigate to the **Platform Tools** folder.

Now type adb devices in the command prompt and hit Enter. If everything works the way it should, you will see your device's serial number under the list of attached devices.

[(base) hafiz@Hafizs-MacBook-Pro mobsf % adb devices List of devices attached 192.168.129.174:5555 device

# How to use Android ADB wirelessly

Once you have set up the ADB and connected the Android device to your PC, follow the steps below to establish a wireless connection to your PC via ADB.

1. Type adb tcpip 5555 in the command line or Termial and press Enter.

```
[(base) hafiz@Hafizs-MacBook-Pro Beetlebug % adb tcpip 5555 restarting in TCP mode port: 5555 (base) hafiz@Hafizs-MacBook-Pro Beetlebug % ■
```

- Find your phone's IP address in Settings > About Phone > Status > IP Address.
- 3. Back in the command line or Terminal, type adb connect [your Android's IP address]
- 4. Press Enter again, the Android device should now be connected

# Genymotion

Genymotion Desktop is an Android emulator you can use to test your Android applications on a wide range of virtual devices for development, test and demonstration purposes.

## **Setting Up Genymotion**

- In order to use Genymotion Desktop, you will need to create and activate
  a Genymotion account. Create your account at
  <a href="https://genymotion.com/account/create">https://genymotion.com/account/create</a>. When prompted for license type,
  select "Personal License" and make sure to click on the verification email
  sent to the email address to activate your account.
- 2. Ensure you have the latest version of VirtualBox installed on your system. Visit <a href="https://www.virtualbox.org/wiki/Downloads">https://www.virtualbox.org/wiki/Downloads</a> to download the latest binary of Virtualbox VM.
- Download and install Genymotion Desktop from https://www.genymotion.com/download/, select your OS
- 4. Launch Genymotion, Sign In and create a new virtual device.
- 5. To install Android applications from the Play Store, you will need to install some additional tools.

# **MobSF**

MobSF framework is an awesome tool for the security analysis of mobile applications. This tool supports both static and dynamic analysis. This section covers MobSF installation on Linux-based distributions (e.g. Ubuntu) and Windows systems.

## **MobSF installation on Linux**

#### Step 1: Download MobSF installer on system

git clone https://github.com/MobSF/Mobile-Security-Framework-MobSF.git

If git not installed on machine,

sudo apt-get install git-all

#### Step 2: Change directory by using the cd command

cd Mobile-Security-Framework-MobSF

### Step 3: Run

./setup.sh

After successful installation. Use the below command to run MobSF.

./run.sh 127.0.0.1:8000

Now you can access MobSF by pointing your browser to http://127.0.0.1:8080

## MobSF installation on Windows

**Step 1**: Download by using the git command or else you can download by browsing URL.

git clone https://github.com/MobSF/Mobile-Security-Framework-MobSF.git

#### Step 2: Change directory to the MobSF directory

cd Mobile-Security-Framework-MobSF

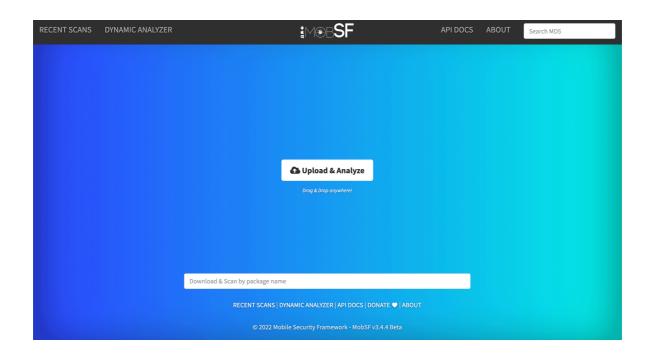
#### Step 3: Install and Run MobSF

setup.bat

#### Step 4: Run MobSF

run.bat 127.0.0.1:8000

Now you can access MobSF using this URL <a href="http://localhost:8000/">http://localhost:8000/</a>



# **JADX**

JADX is a Command line and GUI tools for producing Java source code from Android Dex and APK files.

## **Install on Windows**

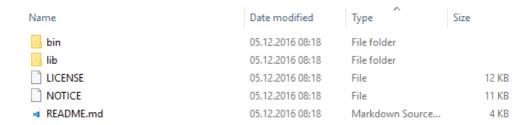
This application is public and licensed under the Apache 2.0 License. The source code is available on Github and the binaries are available to download in the <u>releases window here</u>. Download always the latest version:

#### 1. Download the JADX binaries



#### 2. Extract zip content

Extract the content of the downloaded zip files into a folder of your preference, the content of the zip will be something like:



# 3. Start Jadx-GUI jadx is generally used with a graphical user interface, start the jadx-gui.bat file (inside the jadx-folder/bin folder) with administrator rights.

## Install on macOs

brew install jadx

How to use

## Drozer

Drozer is a security testing framework for Android. It allows you to search for security vulnerabilities in apps and devices by assuming the role of an app and interacting with the Dalvik VM, other apps' IPC endpoints and the underlying OS.

## **Prerequisites**

The following prerequisites must be considered to set up Drozer:

- 1. **Python 2.7** On Windows please ensure that the path to the Python installation and the Scripts folder under the Python installation are added to the PATH environment variable.
- 2. Java Development Kit 1.7

Installation on Windows