Mobile Application Testing Lab

Author: Harshal Harekar
Date: 30/10/2025

Objective

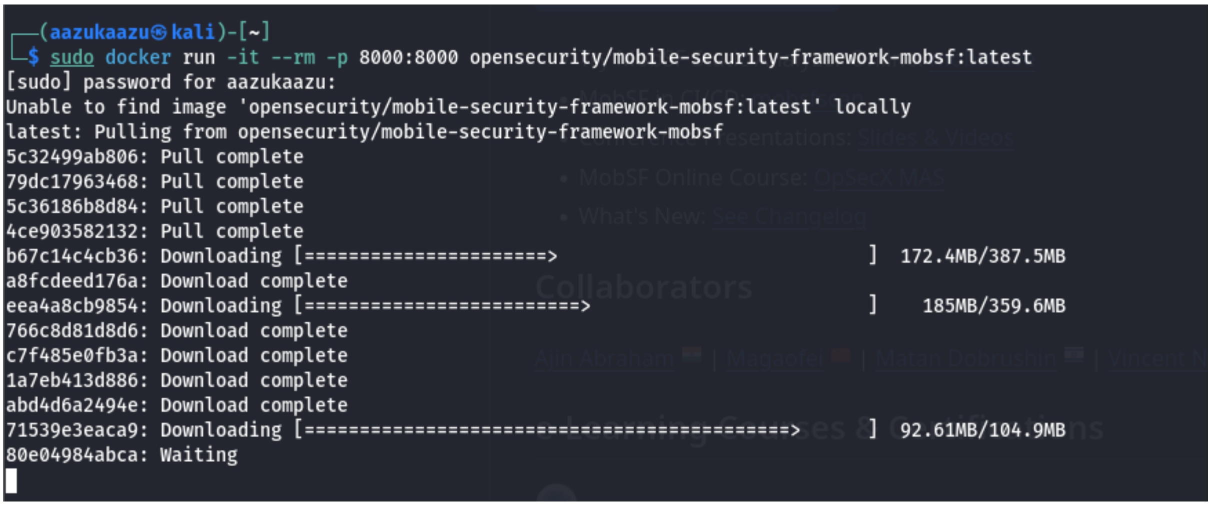
- Static Analysis with MobSF: Identify insecure storage and sensitive data exposure.
- Dynamic Testing with Frida: Hook runtime functions and bypass authentication.
- IPC Testing with Drozer: Discover exposed components and test inter-process communication.

Tools
MobSF, Frida, Drozer
Target APK: DIVA apk

1. Static Analysis with MobSF: Identify insecure storage and sensitive data exposure.

Steps:

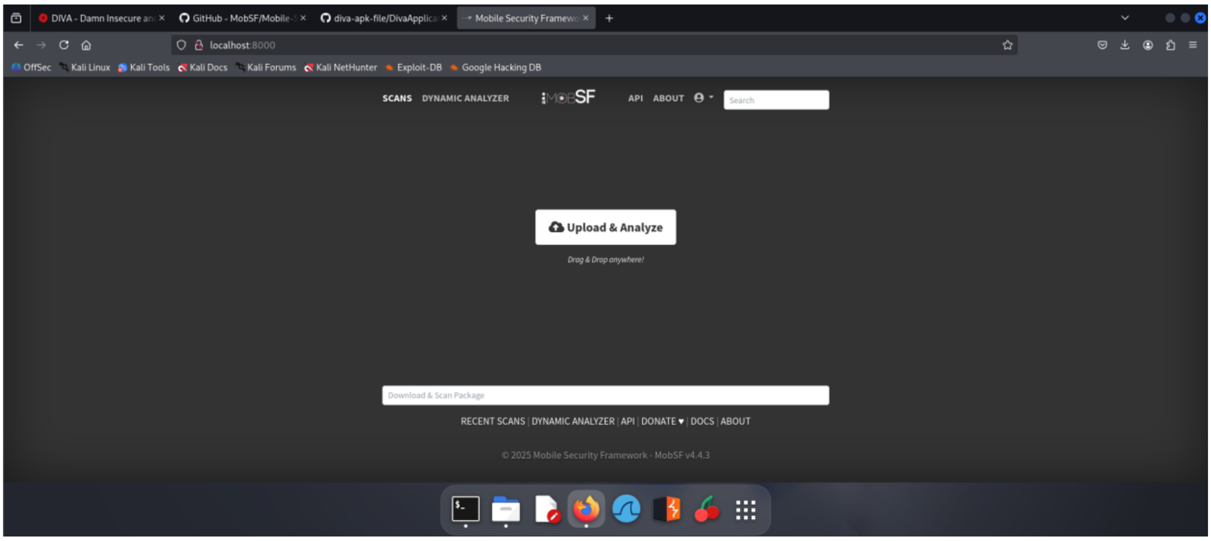
1. Get Target apk like DIVA
2. Launch MobSF
sudo docker run -it --rm -p 8000:8000 opensecurity/mobile-security-framework-mobsf:latest

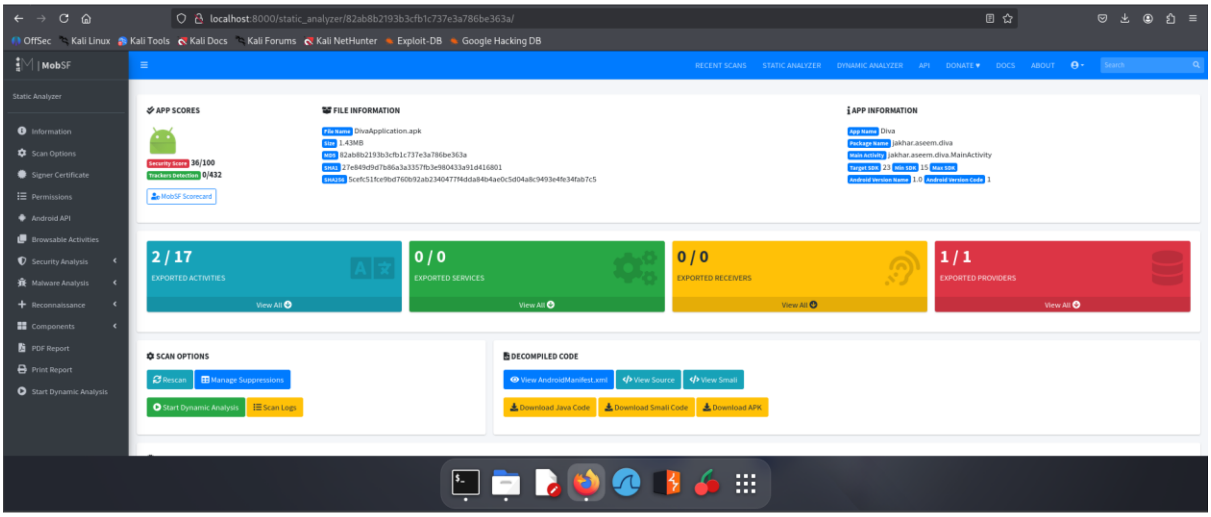


```
┌──(aazukaazu㉿kali)-[~]
└─$ sudo docker run -it --rm -p 8000:8000 opensecurity/mobile-security-framework-mobsf:latest
[sudo] password for aazukaazu:
Unable to find image 'opensecurity/mobile-security-framework-mobsf:latest' locally
latest: Pulling from opensecurity/mobile-security-framework-mobsf
5c32499ab806: Pull complete
79dc17963468: Pull complete
5c36186b8d84: Pull complete
4ce903582132: Pull complete
b67c14c4cb36: Downloading [=====================>                             ]  172.4MB/387.5MB
a8fcdeed176a: Download complete
eea4a8cb9854: Downloading [========================>                          ]    185MB/359.6MB
766c8d81d8d6: Download complete
c7f485e0fb3a: Download complete
1a7eb413d886: Download complete
abd4d6a2494e: Download complete
71539e3eaca9: Downloading [============================================>      ]  92.61MB/104.9MB
80e04984abca: Waiting
```

Once it's running, open your browser and go to *http://localhost:8000*

2. Upload DIVA.apk via the web interface





APP SCORES

Security Score 36/100

Trackers Detection 0/432

FILE INFORMATION

File Name DivaApplication.apk

Size 1.43MB

MD5 82ab8b2193b3cfb1c737e3a786be363a

SHA1 27e849d9d7b86a3a3357fb3e980433a91d416801

SHA256 5cefc51fce9bd760b92ab2340477f4dda84b4ae0c5d04a8c9493e4fe34fab7c5

APP INFORMATION

App Name Diva

Package Name jakhar.aseem.diva

Main Activity jakhar.aseem.diva.MainActivity

Target SDK 23 Min SDK 15 Max SDK

Android Version Name 1.0 Android Version Code 1

2 / 17 EXPORTED ACTIVITIES

0 / 0 EXPORTED SERVICES

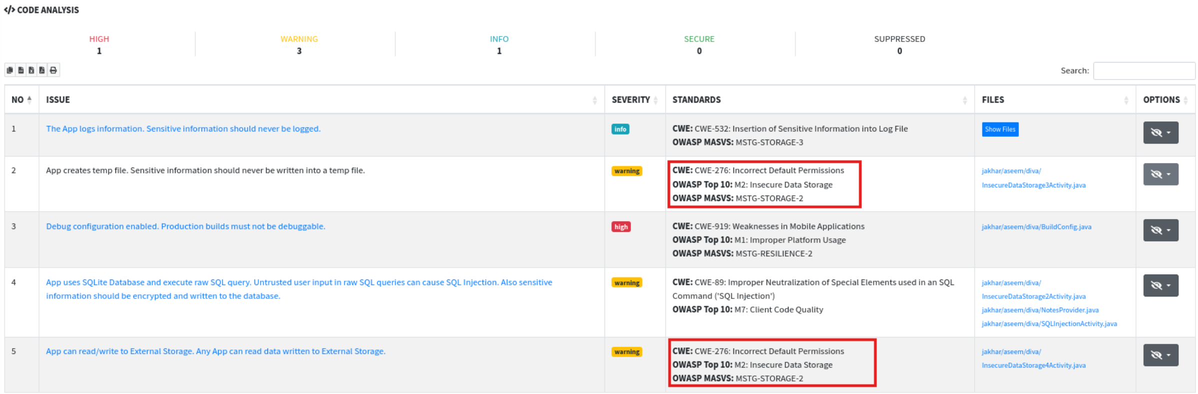
0 / 0 EXPORTED RECEIVERS

1 / 1 EXPORTED PROVIDERS

SCAN OPTIONS

DECOMPILED CODE

3. Review the Security Analysis tab



CODE ANALYSIS

HIGH	WARNING	INFO	SECURE	SUPPRESSED
1	3	1	0	0

Search:

NO	ISSUE	SEVERITY	STANDARDS	FILES	OPTIONS
1	The App logs information. Sensitive information should never be logged.	info	**CWE:** CWE-532: Insertion of Sensitive Information into Log File **OWASP MASVS:** MSTG-STORAGE-3	Show Files	
2	App creates temp file. Sensitive information should never be written into a temp file.	warning	**CWE:** CWE-276: Incorrect Default Permissions **OWASP Top 10:** M2: Insecure Data Storage **OWASP MASVS:** MSTG-STORAGE-2	jakhar/aseem/diva/ InsecureDataStorage3Activity.java	
3	Debug configuration enabled. Production builds must not be debuggable.	high	**CWE:** CWE-919: Weaknesses in Mobile Applications **OWASP Top 10:** M1: Improper Platform Usage **OWASP MASVS:** MSTG-RESILIENCE-2	jakhar/aseem/diva/BuildConfig.java	
4	App uses SQLite Database and execute raw SQL query. Untrusted user input in raw SQL queries can cause SQL Injection. Also sensitive information should be encrypted and written to the database.	warning	**CWE:** CWE-89: Improper Neutralization of Special Elements used in an SQL Command ('SQL Injection') **OWASP Top 10:** M7: Client Code Quality	jakhar/aseem/diva/ InsecureDataStorage2Activity.java jakhar/aseem/diva/NotesProvider.java jakhar/aseem/diva/SQLInjectionActivity.java	
5	App can read/write to External Storage. Any App can read data written to External Storage.	warning	**CWE:** CWE-276: Incorrect Default Permissions **OWASP Top 10:** M2: Insecure Data Storage **OWASP MASVS:** MSTG-STORAGE-2	jakhar/aseem/diva/ InsecureDataStorage4Activity.java	

2. Dynamic Testing with Frida: Hook runtime functions and bypass authentication.

Steps

1. Setup Frida

```
pip install frida-tools
```

2. On Android Emulator
Install Frida server:

```
adb push frida-server /data/local/tmp/
adb shell "chmod 755 /data/local/tmp/frida-server"
adb shell "/data/local/tmp/frida-server &"
```

3. Hook

```
frida -U -n DivaApplication.apk
```

4. Inject Script
JavaScript.js:

```
Java.perform(function () {
  var Login = Java.use("com.testapp.LoginActivity");
  Login.checkPassword.implementation = function (input) {
    return true;
  };
});
```

3. IPC Testing with Drozer: Discover exposed components and test inter-process communication

Steps

1. Install Drozer

```
apt install drozer
```

2. On Android Emulator
Install Drozer agent APK
Start agent
Connect:

```
adb forward tcp:31415 tcp:31415
drozer console connect
```

3. Scan for IPC issues

```
run app.activity.info -a com.testapp
run app.broadcast.info -a com.testapp
```

4. Exploit

```
run app.activity.start --component com.testapp/.LoginActivity
```

Log

Test ID	Vulnerability	Severity	Target App
1	Insecure Storage	High	DivaApplication.apk
2	Auth Bypass (Frida)	Critical	DivaApplication.apk
3	Exported Receiver	Medium	DivaApplication.apk