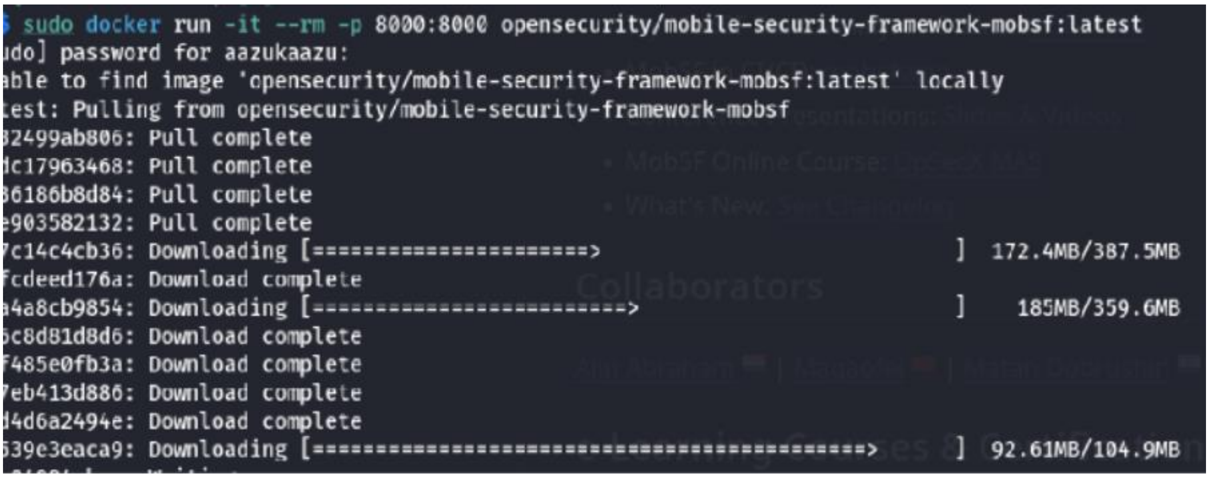
1. Mobile Application Testing

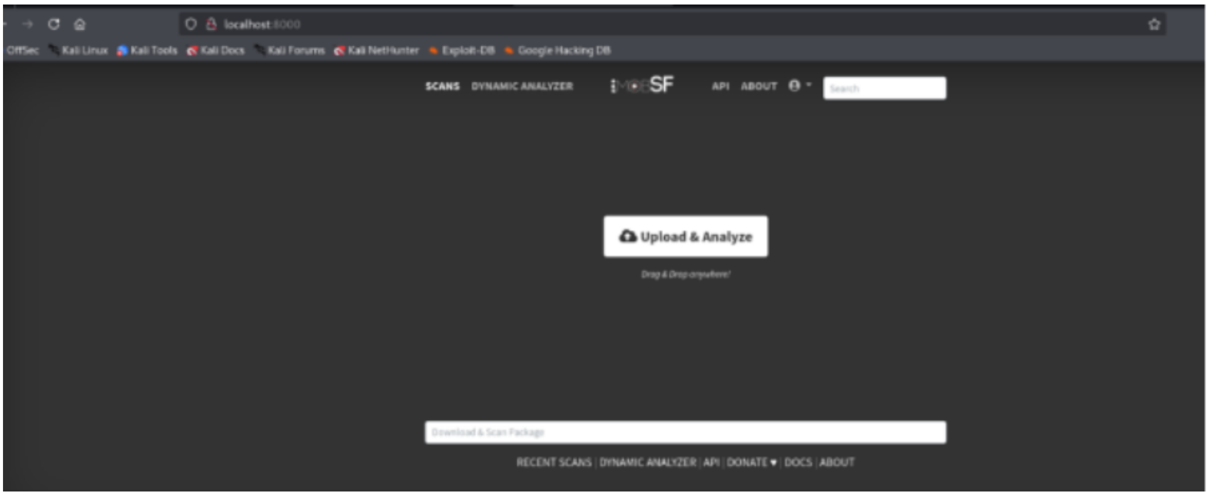
1.1 Static analysis with MobSF

First, setup the target as DIVA apk and launch the MobSF for static analysis

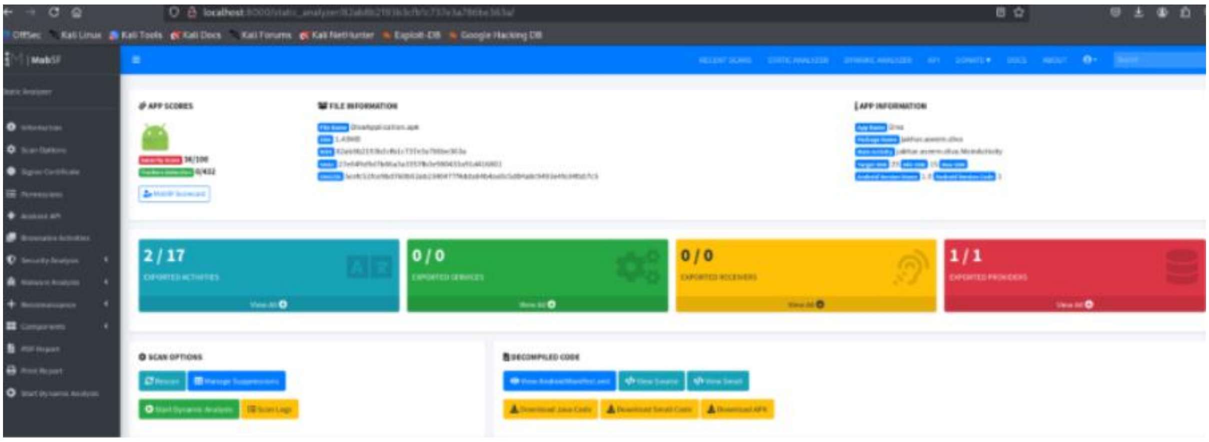


```
$ sudo docker run -it --rm -p 8000:8000 opensecurity/mobile-security-framework-mobsf:latest
udo] password for aazukaazu:
able to find image 'opensecurity/mobile-security-framework-mobsf:latest' locally
test: Pulling from opensecurity/mobile-security-framework-mobsf
32499ab806: Pull complete
1c17963468: Pull complete
36186b8d84: Pull complete
903582132: Pull complete
7c14c4cb36: Downloading [======================>                           ]  172.4MB/387.5MB
fcdeed176a: Download complete
a4a8cb9854: Downloading [=========================>                        ]    185MB/359.6MB
5c8d81d8d6: Download complete
f485e0fb3a: Download complete
7eb413d886: Download complete
14d6a2494e: Download complete
539e3eaca9: Downloading [===========================================>      ]  92.61MB/104.9MB
```

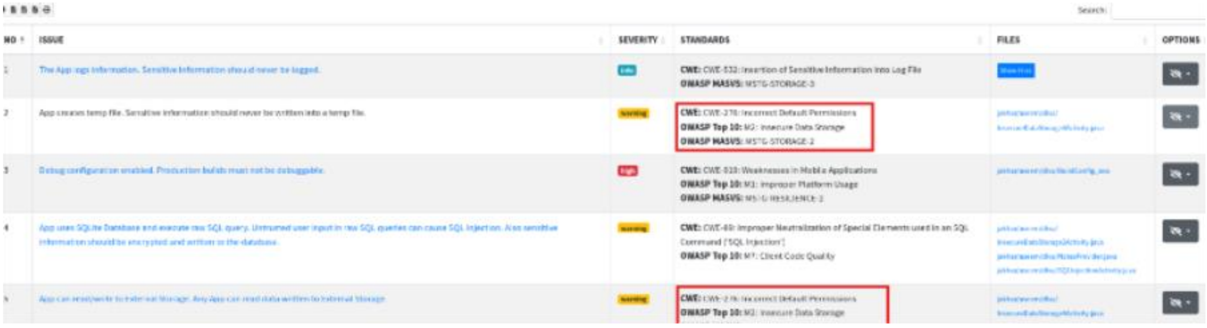
Once, it was running open the browser and go to http://127.0.0.1:8000



Upload a DIVA file in the web server, it will show some results.



Then, you have to analyse the security Assessment lab



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	jakhar/aseem/diva/ InsecureDataStorage4Activity.java	