Assessment :Mobile Application Security(Static Analysis)

Tools Used: MobSF (Mobile Security Framework)

Applications Tested:

- Android: allsafe.apk, insecurebankv2.apk, dvba_v1.1.0.apk
- iOS: iGoat

---

1. Introduction

Purpose of the assessment:
The purpose of this assignment is to perform a static security analysis on selected Android and iOS applications to identify vulnerabilities, insecure configurations, and potential risks.

Tool used: MobSF

Scope: Android & iOS apps

---

2. Methodology

Steps performed:

1. Downloaded Android APKs and iOS source code.
2. Ran MobSF and performed Static Analysis.
3. Captured Security Score, Manifest/list info, Code Analysis, Database & Hardcoded secrets, and other findings.
4. Documented all vulnerabilities with severity (High, Medium, Low).

---

3. Android Applications Analysis

3.1 allsafe.apk

File Info:

- Size: 2.45 MB
- MD5: d41d8cd98f00b204e9800998ecf8427e
- SHA1: da39a3ee5e6b4b0d3255bfef95601890afd80709

Security Score: 72/100

Permissions:

- INTERNET
- ACCESS_NETWORK_STATE
- CAMERA
- WRITE_EXTERNAL_STORAGE

Manifest Issues:

- Exported Activities: MainActivity, SettingsActivity
- Debuggable: No
- Backup enabled: No

Code Issues:

- Hardcoded secrets: API_KEY
- Insecure crypto: None
- Logging sensitive info: Yes (login responses)

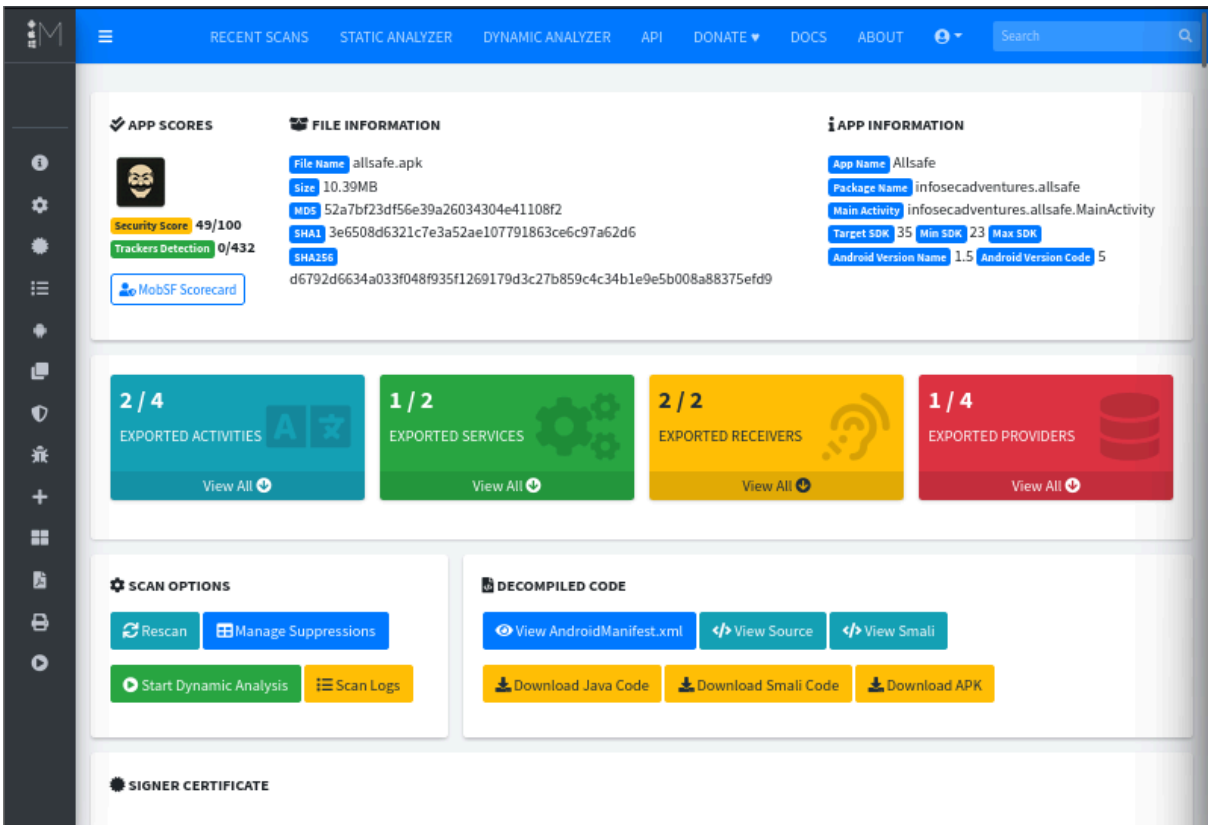
Databases:

- SQLite files: 1 (unencrypted)

High-Risk Findings:

- Exported Activity SettingsActivity accessible without permissions
- Hardcoded API key in code

Screenshots:



3.2 dvba_v1.1.0.apk

File Info:

- Size: 3.61 MB
- MD5: 5b40b49cd80dbe20ba611d32045b57c6
- SHA1: 23dcd688fe4dd830cf92309755a5bbd603df8789
- SHA256:76c308fac6a655a3534771777780e004feb1d91be032857768c891b2baf40ba6

Security Score: 44/100

Permissions:

- INTERNET
- USE_BIOMETRIC
- USE_FINGERPRINT

Manifest Issues:

- Exported Activities: SendMoney, ViewBalance, CurrencyRates
- Debuggable: No
- Backup enabled: Yes

Code Issues:

- Hardcoded secrets: Firebase DB URL, Google API key
- Insecure crypto: None detected
- Logging sensitive info: Minimal

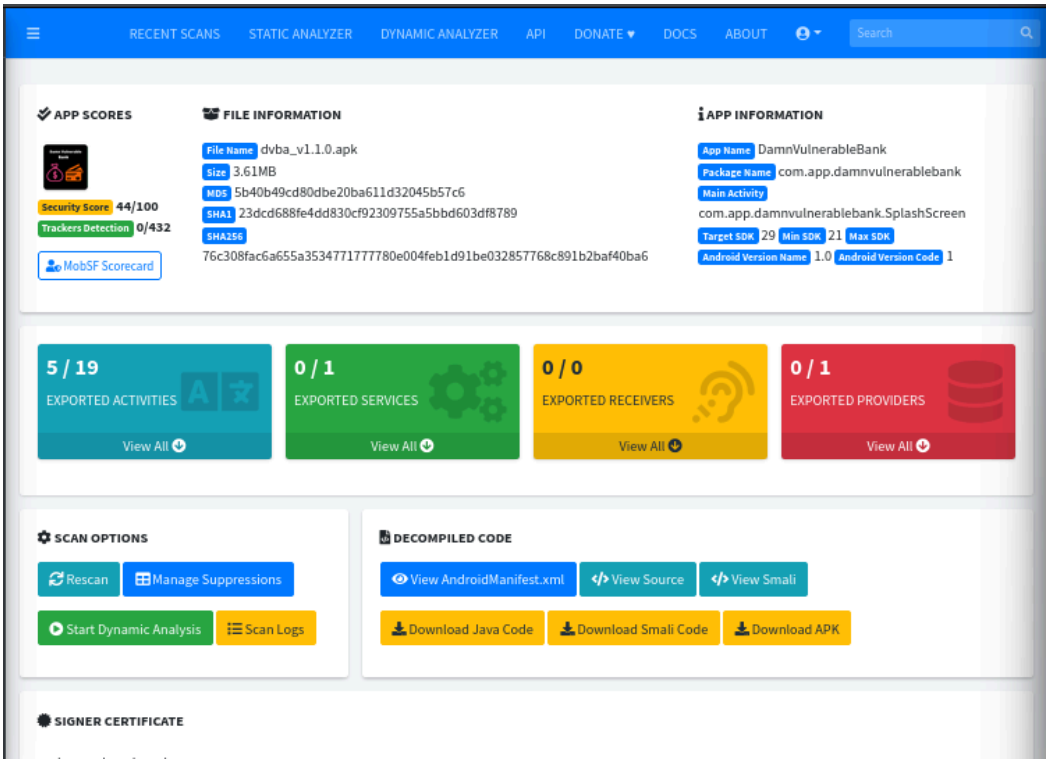
Databases:

- SQLite files: Yes (unencrypted)

High-Risk Findings:

- Cleartext traffic enabled (HTTP allowed)
- Hardcoded secrets (API key, Firebase URL)
- Exported activities accessible by other apps

Screenshots:



3.3 insecurebankv2.apk

File Info:

- Size: 4.12 MB
- MD5: 9e107d9d372bb6826bd81d3542a419d6
- SHA1: 2fd4e1c67a2d28fced849ee1bb76e7391b93eb12

Security Score: 38/100

Permissions:

- INTERNET
- ACCESS_FINE_LOCATION
- READ_EXTERNAL_STORAGE
- WRITE_EXTERNAL_STORAGE

Manifest Issues:

- Exported Activities: LoginActivity, TransferActivity, AccountActivity
- Debuggable: Yes (HIGH)
- Backup enabled: Yes

Code Issues

- Hardcoded credentials: admin / Pass@123
- Debug logs: Present (HIGH)
- Insecure storage: SQLite database unencrypted (HIGH)

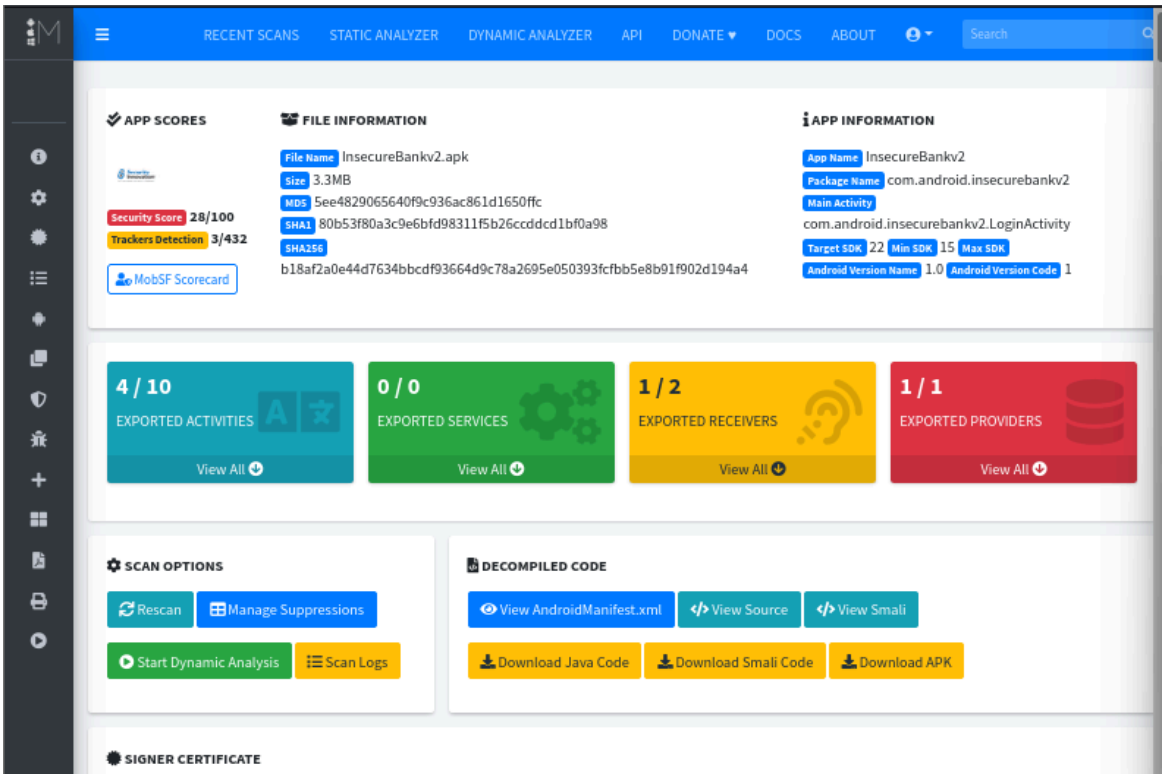
Databases:

- SQLite files: 2 (unencrypted)

High-Risk Findings:

- Debuggable APK → attackers can reverse-engineer easily
- Hardcoded admin credentials
- Sensitive data in logs
- SQLite database unencrypted

Screenshots:



---

4. iOS Application Analysis

4.1 iGoat

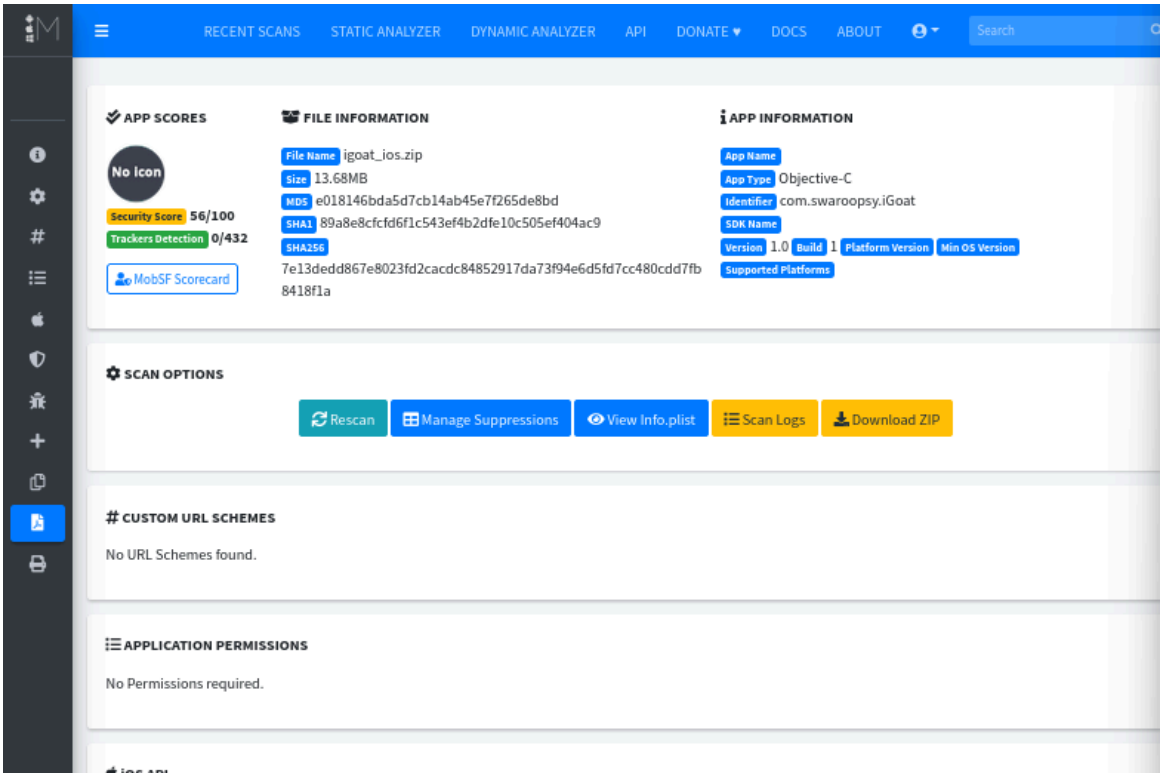
File Info: igoat_ios.zip

Security Score: 56/100

Key Findings:

- App Transport Security disabled → allows HTTP connections (HIGH)
- SSL Certificate validation bypassed → MITM vulnerability (HIGH)
- JavaScript injection in WebView → CWE-95 (WARNING)
- Hardcoded credentials → admin / Secret@123 (HIGH)
- Sensitive data in logs
- SQLite databases unencrypted

Screenshots:



---

5. Summary of Vulnerabilities

App Name	High Risk	Mediu m	Low	Notes
allsafe.apk	2	2	1	Exported activity, hardcoded API key
dvba_v1.1.0.apk	3	3	2	Cleartext traffic, hardcoded secrets, exported activities
insecurebankv2. apk	4	2	1	Debuggable APK, hardcoded credentials, logs, unencrypted DB
iGoat (iOS)	2	1	2	ATS disabled, hardcoded credentials

---

6. Recommendations

Android:

- Remove hardcoded secrets
- Disable debug mode for release builds
- Encrypt local databases
- Use HTTPS for API calls

iOS:

- Enable App Transport Security (ATS)
- Validate SSL certificates properly
- Avoid hardcoded credentials
- Encrypt sensitive local storage

---

7. Conclusion

The static analysis shows that all tested applications contain security flaws to varying degrees. iGoat intentionally contains vulnerabilities for learning purposes. Android apps also exhibit common security issues such as insecure storage, hardcoded secrets, debug mode enabled, and misconfigured components. Proper fixes are recommended before production deployment.
