Project Phase II

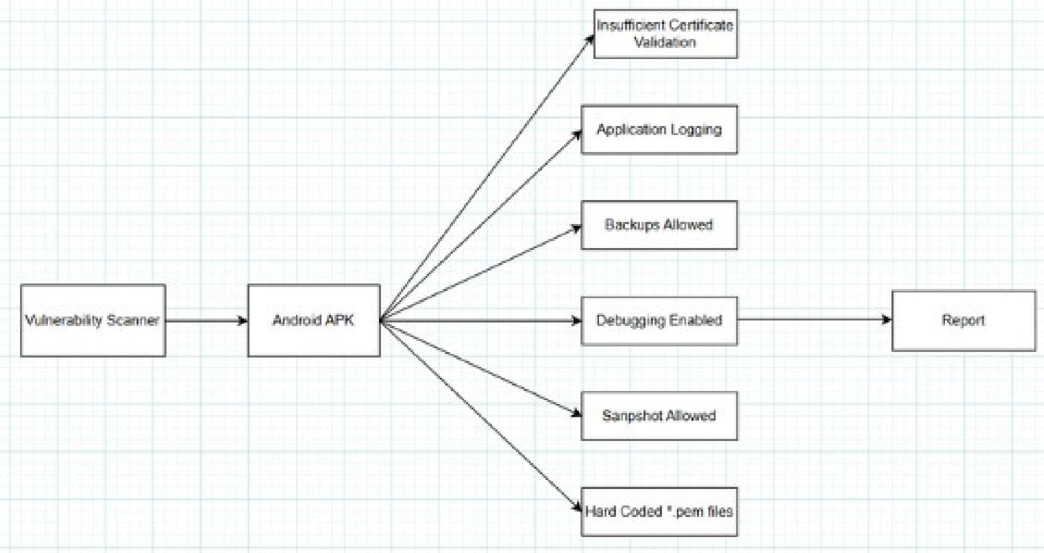
Enhance Android APK Vulnerability Detection And Exploitation

Motivation

- Modern mobile applications often contain vulnerabilities that can lead to data breaches, unauthorized access, and privilege escalation.
- Traditional static analysis tools detect vulnerabilities but do not show how to exploit them. Many penetration testers and security analysts are left with false positives or unverified risks.
- We go beyond static analysis by automating exploitation attempts, making it a practical pentesting tool.

Problem Statement

- Existing static analysis tools for Android APKs often produce false positives and lack practical exploit guidance, making it difficult for security professionals to validate and demonstrate real-world vulnerabilities effectively.
- To bridge the gap between vulnerability detection and practical exploitation by integrating tools like Metasploit and providing an intentionally vulnerable APK (exploit.apk) for real-world validation and demonstration of identified security risks.



Vulnerability Scanner

Android APK

Insufficient Certificate Validation

Application Logging

Backups Allowed

Debugging Enabled

Sanpshot Allowed

Hard Coded *.pem files

Report

Methodology

- Perform static analysis on Android APK files using tools like apktool and jadx to extract and inspect code and configurations.
- Identify security misconfigurations and vulnerabilities such as logging enabled, cleartext traffic, debugging flags, and hardcoded secrets.
- Provide detailed explanations and command-line examples to demonstrate potential exploitation, with optional integration of Metasploit for real attacks.

Outcome of project

- Users gain detailed explanations for each detected issue, enabling better comprehension, faster remediation, and accurate reporting.
- Through verbose mode, users gain detailed explanations for each detected issue, enabling better comprehension, faster remediation, and accurate reporting.