```
┌──────────┐     ┌──────────┐     ┌──────────┐     ┌──────────┐
│ Pre-sale │ ──> │ Pentest  │ ──> │Documents │ ──> │   ...    │
│          │     │          │     │and Report│     │          │
└──────────┘     └──────────┘     └──────────┘     └──────────┘
```

0. Nhận APK/IPA

1. Recon → **Application Metadata** → • Check via MobSF → **Document to Sheet**

Toosl: JEB, Byte Code Viewer, apktool, dex2jar, jd-gui

Decomplie app to Readable form ← Use for Static Analysis ← **2. Set up env**

2. Set up env → Use for Dynamic Analysis → **Bypass Detection**
- Emulator Detection
- Root / Jailbreak Detection
- SSL Pinning

How-to →
- Tools: Magic mask, Objection, Frida, RootCloak.
- Modify Code => Rebuild app

Note

>20% time = Ask for uncheck version (No / Yes)

3. Pentest Mobile Features and Backend API

M3 → **Match the Features with API endpoint** → **Document to Sheet**

→ **Pentest API Endpoint (references API:2019 OWASP)**

M6 → **Test for communication Channel**
- Data transmission over HTTP
- Authentication over HTTP

In-use Data — M2 → **Local Data Storage**
- Local data Storage
- Logs
- Snapshots
- Backups

4. Pentest Mobile Client

M5 → **Cryptography**
- Hardcoded secret keys
- Weak cryptographic algorithms
- Random Generator

M1 → **Platform Interaction**
- App Permissions
- Injection Flaws
- URL Schemes
- WebViews -> XSS, file access

review source code

M4 → **Security Configuration (Code Quality)**
- Signing information
- Transport Security Configuration (iOS - ATS)
- Default Security Features (PIE, ARC, etc)

M7 → **Code Tampering & Reverse Engineering**
- Device Binding
- File Integrity

5. Document

Vulnerabiltiy:
- Executive summary
- Scope & Context
- Methods used
- Sources of info
- Prioritized finding
- Detailed Finding Recommendations