Android Security Course

Prerequisites for the Android Security Course

To effectively participate in this course, students should have **Basic Shell Command Knowledge**:
Familiarity with Linux shell commands, particularly in **Bash**, as they are essential for working with **ADB (Android Debug Bridge)** and analyzing Android applications.

Android OS

- The Android Open Source Project
- Google mobile services
- Platform Architecture (Kernel, HAL, ART, NDK, Java and Kotlin API Framework, System Apps)
- Android Core - Virtual Machines: JVM, DVM & ART
- Android OS boot process explained with focus on Zygote
- Processes & Threads
- Android Memory Management
- Android File System Overview
- Security: Device, Kernel and App
- Android System permissions (signature|privileged)
- Android Platforms (Wear, TV, Android for Cars)
- Android releases primary key features with emphasis on security enhancements
- AndroidX support library and Android Jetpack

Android Studio & Developer Signature

- Android Studio Overview
- Projects overview (App Modules, App files and directory structure)
- Build and run your app and Build Configurations
- App Gradle
- App Profiling and performance
- Analyze your build with app analyzer
- APK Signing using private and public keys
- Detecting apk developer signature and locating apps with same signature

Command line tools

- Android SDK tools
- **Android Debug Bridge(ADB)**
  - Connecting to devices
  - Installing and locating APK
  - copying files from/to device
  - shell commands (Activity Manager, broadcasts, services)
  - Querying device (RAM, installed apps and more)

  - Detect apps anomaly behavior
  - Detect App excessive device usage (RAM, battery, sensors and other hardware)
  - Key events
  - Extracting files from apk

Android Application

- App Components (Activity, Service, Broadcast Receiver & Provider)
- App states (Foreground, Background, idle)
- Intents and Intent Filter - communication between processes
- App Manifest File
- App Security & Permissions
- In-depth investigation app's Manifest file (Inc. live demonstrations)
- Creating Deep links into Apps employment and protections
- User Location & Google play services
- Foreground & Bounded Services
- App Storage (internal & external)
- File Provider
- Multi-Threading app
- Summery - Security, Processes & IPC (Inter Process Communication)

From theory to practice - Malicious app demonstration

- Intro to ASR (Automatic Speech Recognition)
- Misleading the user to give us dangerous permission
- Creating a background service hidden from the user with non-stop server communication
- Hiding this service from the end user while adopting the newest OS security features
- Receiving System events like the end of boot process and keeping our service "alive" indefinitely
- Hiding our activity from the user using system events like screen lock, changing of simple user audio settings and call settings
- Storing processed data on cloud, database security and user association
- How to reveal this app from Manifest file
- Is it even remote possible on iOS?

Android Vulnerability aspects and detecting Malicious software

- Creating a backdoor for outside app communication
- Dynamic loading and execution of external app code
- Exploring apps hard-coded values (like strings) and detecting malware
- Preform APK reverse engineering to project and extract files (Both Kotlin and Native cpp code)
- Common Encoding (Base64, Hex) and Encryprion (XOR, RC$, AES) methods.

- Code scrambling with D8 & Pro-Guard

Intercepting and Analyzing Web/API Traffic from mobile app - Man In The Middle Attack

- Start Burp Suite and Configure the Proxy
- Configure Android to Use the Burp Proxy
- Installing the CA Certificate using ADB
- Understanding HTTPS Interception via CA Certificates
- Capturing API Mobile app Traffic
- Code Example of a Vulnerable API Request
- Bypassing HTTPS Interception Using Android’s Network Security Config
- Secure App Version with Certificate Pinning

Schedule

Day 1

Android OS

Mostly **Theory** about 45 min practice in activating system services

Day 2

part 1 - **Android Studio** and Developer signature (Theory + Practice)

part 2 - **Android Debug Bridge** full capabilities (Practice)

Day 3

Part 1 - **Android Application** components and Manifest file (Theory + Practice)

Part 2 - Malicious Spy App Demonstration (Practice)

Day 4

Part 1 - **Android Vulnerabilities** (Theory + Practice)

Part 2 - Intercepting web traffic (Theory + Practice)