Project synopsis:

Short Title: Android/iOS App Security

Title: Covert channel communication on Android OS / iOS

Project lead and contact details: Mehmet Belviranli, belviranli@mines.edu

Suggested team size: 4-5

Logistics: On-campus.

Project description:

Android and iOS apps require explicit user permissions for every component in the phone they want to access and use. A recent study uncovered an exploit where applications on mobile platforms reveal signatures that can be observed by other applications in the system.

In this project, the team is expected to create two separate Android and iOS app pairs that will communicate with each other via the exploit explained above. There will be two applications with different permissions (e.g. contacts access and health sensor access) and these two applications will send each other the data that they are not supposed to have access to. The communication will not be over internet or any conventional means (e.g., interprocess communication). Instead, the communication will be over the memory-based exploit. An incomplete template for the Android application pairs will be provided.

Project components:

The project requires multiple components to be implemented on the Android and iOS platform:

Covert communication protocol: The covert communication protocol will be used internally by the applications below.

Applications: The two applications should carry a basic functionality (i.e. display contacts or a widget that displays the number of steps taken that day) which would justify their permission request. Applications will look innocent, and their malicious behavior will not be detectable.

Desired skills:

Following skills are essential but not necessarily required:

- Familiarity with Android/iOS programming environment.
- Familiarity with C bridge in android programs
- General computer organization knowledge (i.e., caches, cores, memory etc.)

Devices available:

- Qualcomm Snapdragon 865 development board
- NVIDIA Xavier and Orin series Socs.

Expected Outcome:

At the end of the project, the team is expected make a demo showing that the two applications communicate with each other (by not using conventional means, such as internet or IPC.)