13.3.4 Mobile Device Operating System Facts

Two operating systems dominate the smart phone market, Android and iOS.

This lesson covers the following topics:

- Open-source Android
- Android stack
- The iOS operating system
- The iOS stack
- Android, iOS, and the Mobile Security Model
- Rooting and jailbreaking

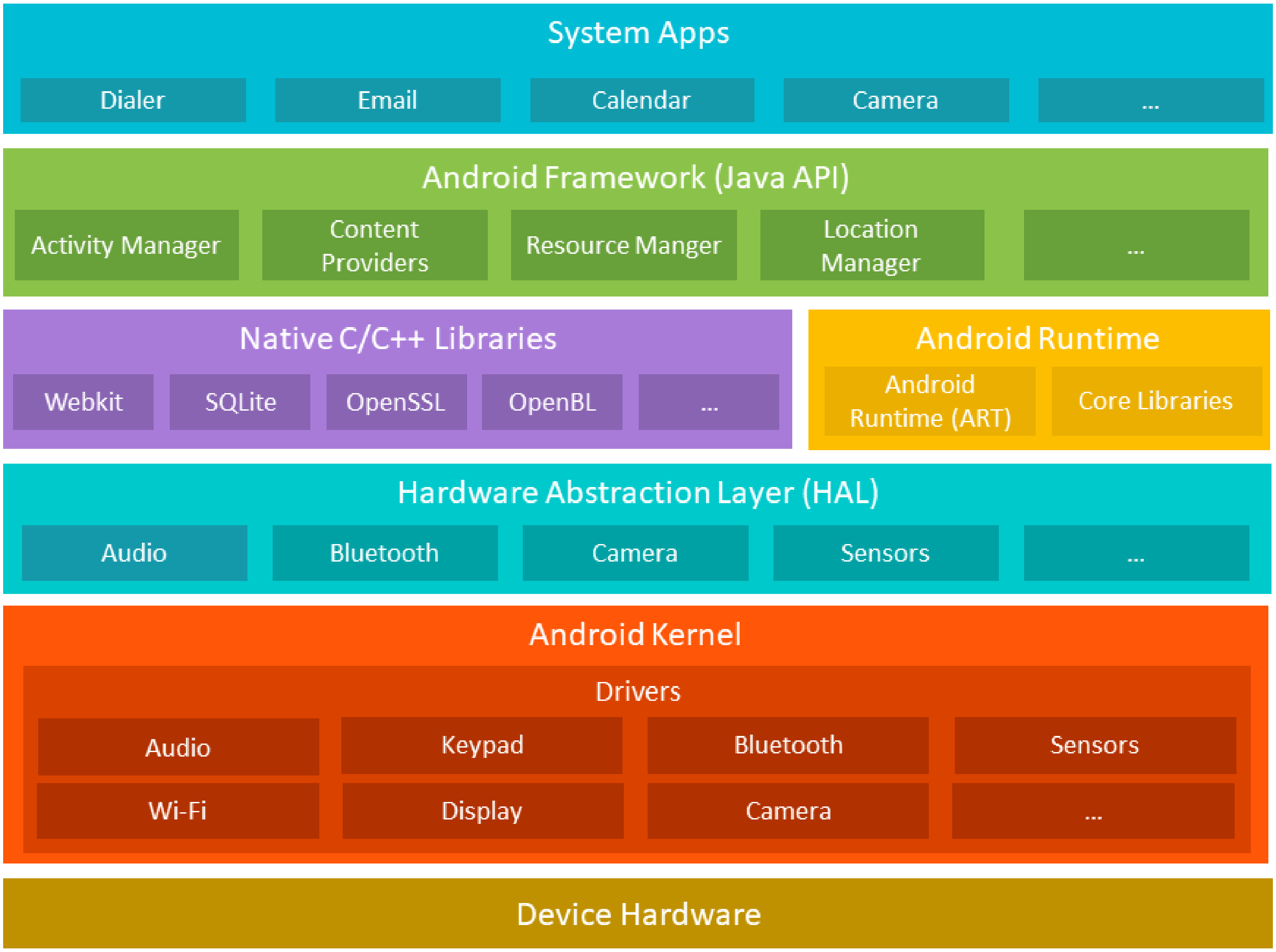
Open-Source Android

The Android mobile operating system was developed by Google. Key features are:

- Android is an open-source platform that is based on a Linux kernel.
- Android source code is maintained by the Android Open Source Project (AOSP). Android source code:
  - Is stored in a Git repository
  - Is available to anyone
  - Can be redistributed and modified
- Mobile device manufacturers modify the open-source Android source code to create images that are installed on their mobile devices.

Android Stack

The Android components can be visualized by arranging them in a stack. Each layer in the stack interfaces with only the layer above and below it.



The following table describes a few important Android components.

Android Component	Description
Device hardware	Technically, a mobile device's hardware isn't part of the Android operating system. - Android can run on different hardware configurations such as smart phones and tablets. - Mobile device manufacturers differentiate themselves by offering different hardware. - Manufacturers add drivers to the kernel that interface with the device hardware. - The Hardware Abstraction Layer (HAL) is related to the device hardware component. The HAL is mainly maintained by the AOSP, but manufacturers add routines to the HAL to interface with their device drivers.
Android kernel	Often, the kernel is described as the Android operating system, and the whole stack is called the Android platform. - The Android kernel is built from the Linux kernel. - The kernel accesses the device's hardware resources such as the touch screen, camera, and GPS sensor. - The kernel receives input from and sends data to the HAL.
Android Runtime	The Android Runtime (ART) component is built specifically for Android. - ART was created to address battery life and processing power problems. - ART provides a common runtime environment for applications. - Applications are written in Java. - ART provides the Java runtime environment. - Older Android versions used just-in-time compilation, which compiles the Java just before the app is used. - ART provides ahead-of-time compilation by compiling the application when it's first installed.
Android applications	Android applications extend the features of the Android platform. - A few core apps (phone dialer, email client, calendar, and camera) are provided in the AOSP code. - Mobile device manufacturers distinguish themselves by modifying the AOSP apps or a set of pre-installed apps.
Google cloud-based services	Google provides cloud services to any Android-compatible device. - Google Play allows users to discover, install, and purchase apps from their Android devices. - Android Update delivers security updates and new Android OS features to Android devices. - Google offers a framework that allows Android applications to use cloud capabilities to back up data and settings and use cloud-to-device messaging.

The iOS Operating System

The iOS operating system powers Apple mobile devices such as the iPhone, iPod touch, and iPad. As a contrast to the open-source Android platform, iOS is a proprietary operating system that runs only on Apple hardware. An advantage is that iOS is more secure. A disadvantage is that lower-level workings are not as well documented.

The iOS Stack

The iOS operating system is advertised as a trimmed-down version of OS X for Mac and can be visualized as a layered stack. Remember:

- Below the application layer are the layers that make up the true iOS operating system.
- Each layer offers its own set of technologies or frameworks.
- Upper-layer frameworks are built on lower-layer frameworks.
- Applications can interface directly with the frameworks offered by each layer instead of going through intermediate frameworks.

The following table lists the iOS layers:

iOS Layer	Description
Application	Apple offers its own system apps. Once third-party apps are registered with Apple, they can be installed from Apple's App Store.
Core OS (iOS kernel)	The Core OS layer contains the low-level frameworks like security, networking, and the file system.
Core service	The Core Services layer contains public and private frameworks. Public frameworks (like contacts, location, and image) can be used by third-party apps. Private frameworks (like SMS, phone, and calendar) are intended for only Apple's apps.

Media	The media layer contains the graphics, audio, and video frameworks for multimedia experiences.
Cocoa Touch	Applications can reach down to any lower frameworks. However, most applications use the frameworks offered by the Cocoa Touch layer. For example, Cocoa Touch framesworks: ▪ Define the app's appearance, the use of multitasking, and touch-based input. ▪ Contain the UIKit developers use while creating new apps.

Android, iOS, and the Mobile Security Model

The Android operating system covers all five core areas of the mobile security model. The iOS operating system covers only four.

Mobile Security Model Areas	Covered by Android	Covered by iOS
Traditional access controls	Yes	Yes
Digital signing	Yes	Yes
Encryption	Yes	Yes
Isolation	Yes	Yes
Permissions-based access controls	Yes	No

Rooting and Jailbreaking

Some of the areas in the security model followed by Android and iOS can be overridden. You can override them by rooting an Android device or jailbreaking an iOS device.

- Rooting and jailbreaking overcomes the security restrictions imposed by the device's manufacturer to:
  - Modify, remove, or replace applications.
  - Sideload unapproved applications.
  - Run apps with administrator or root privileges.
  - Change system settings.
  - Gain low-level access to hardware.
  - Modify or delete system files.
  - Replace the operating system.
- Android devices are rooted to:
  - Visually change the appearance or theme.
  - Increase performance by overclocking the CPU or GPU.
  - Remove bloatware that comes pre-installed on their device.
- Android devices are rooted:
  - By exploiting vulnerabilities in the device's firmware to copy the su binary to a location in the process's PATH.
  - By using the chmod command to grant the su binary execute permissions.
  - With rooting tools, including KingoROOT and TunesGo One-Click Root Android.
- Android devices aren't rooted to allow application sideloading. While it may give you a warning, Android allows you to install applications outside of the manufacturer's app repository or outside of the trusted Google Play Store.
- iOS devices are jailbroken to sideload or install applications that are not found in Apple's approved App Store.
- iOS devices are jailbroken:
  - By installing kernel patches that have been modified to allow you to load trusted third-party applications.
  - With jailbreaking tools, including Cydia, Yalu, Velonzy, and Keen Jailbreak.
- Rooting or jailbreaking a mobile device may:
  - Void the device's warranty.
  - Cause poor performance.
  - Incur malware infections.
  - Brick the device, making it impossible to turn on or repair.