

Sign iOS applications for Gradle Plugin

Apple requires that applications be signed to run on an Apple device. This code signing "assures users that it is from a known source and the app hasn't been modified since it was last signed." In addition, Apple distributes two different types of Apple ID Certificates used to sign the applications:

- **Developer Profile** - this certificate is registered for development and testing of the application on a registered set of devices, usually in-house.
- **Distribution Profile** - this certificate is used by developers wishing to distribute applications through the Apple Store. Applications signed with this certificate may run on any device, after verification.

There are two scenarios for running applications using Perfecto Lab devices:

- **Appium/Perfecto Native Automation (or similar) automation scripts** - These automation scripts automate a "distribution" application, installed on the Perfecto Lab device. The code is signed with the owner's distribution certificate, or, if there is a need to [instrument](#) the application, with a Perfecto certificate.
- **XCTest/XCUI Test unit/UI tests** - These test methods are executed as part of the development process, in a development mode. To execute them on a Perfecto Lab device, there is a need to sign them with a **Developer ID** that includes registration of the *device-id* as part of the developer certificate provisioning file.

Provisioning file

As part of the signing process of Apple applications, a *Provisioning Profile* file is generated and attached to the ipa file. The provisioning profile includes information on the feature entitlements (for example, iCloud, push notifications) used by the application. There are three types of profiles:

- **Development** provisioning - allows you to test your application in debug-mode. This allows execution of both XCTest and XCUI Test tests.
- **App Store Distribution** provisioning - the final distribution profile provided to the end-users that download the application from the App Store.
- **Ad-hoc Distribution** provisioning - Used for building the application for beta testers, does not allow debugging the application.
- **Enterprise Distribution** provisioning - Specially licensed distribution for non-App Store distribution, to in-house devices, does not support debugging the application.

For testing the application with the Gradle Plugin we require a **development** provisioning file.

When using the Gradle Plugin to execute XCTest unit tests, or XCUI Test UI tests on a real Perfecto Lab device, the *ipa* files (both the application/XCTest *ipa* file and the XCUI Test Runner *ipa* file) must be signed with a Developer Certificate that includes a provisioning file with the device's UDID provisioned.

The Gradle plugin supports the following options, selected with the **resign** Pre-Execution configuration parameter:

- Automatic Perfecto Signing - Perfecto will replace the files' signature with Perfecto's Apple Developer ID, with the device provisioned. However, any entitlements may be removed. To select this option (default) - the **resign** parameter should be set to **true**.
- No Resigning - Customer registers the Perfecto Lab devices with their Developer certificate, sign the application and Runner *ipa* files with this certificate, add the Perfecto device's UDID to the provisioning profile, and use the Gradle plugin to execute on the device. To select this option - the **resign** parameter should be set to **false**.