iOS Troubleshooting - Install App Fails

Issue: App fails during installation process and following error is received

```
installApplication command failed. Reason: handset server: Cannot install
application. file is not signed properly or has the wrong provisioning
'Install' completed with errors
```

Interactive and Automation Scenarios:

You are installing the app using the widget on the Lab interactive page or using an Automation script and no form of instrumentation is selected.

Solution:

Your application has a file inside it called the embedded.mobileprovision file. This file contains all the UDID (device ID's) for your iOS devices.
The application developer needs to take the device UDID's sign the embedded.mobileprovision file with them.
You can manually check the application embedded provisioning file to make sure the device UDID is listed.

How to check if an UDID is added to the app embedded provisioning file

1. Download and unzip the <app-name>.ipa file on your computer
2. Open the <app-name>.app directory
3. Find the file called "embedded.mobileprovision" and open it with Notepad++ or any text editor
4. Look for the string <key>ProvisionedDevices</key> which will list all the device UDIDs.
5. If the above key is missing, look for a key: <key>ProvisionsAllDevices</key>
   1. If the key exists and the value is *true*, then the app can be installed on any device. (Enrolled iOS Developers in the Enterprise program have the ability to distribute their in-house applications without identifying individual devices or submitting the application to the App Store.)

XCUITest/XCTest Scenarios:

Scenario 1 : You used R*esign=false* in the configuration file

This likely means that the devices are not registered in customer provision profile.

Solution: Use the *parameter = true* (set by default) and then the app will be resigned with the Perfecto developer signature.
Note: Entitlements may be removed.

Please see more about this here: Signing iOS Applications

Scenario 2: Application is signed with different signature from the one that used for installation

Ideally, customer dev teams would push the ipa/ipa-runner to Perfecto as per their build process because is the only sure way we know that the QA team is testing the proper app version and the signature is correct. (For example, if the signature for latest app version is different then the one used for installation and testing.)

In this scenario, and when using DevTunnel, the provision profiles must be installed by the customer on devices. Usually this is performed by developer teams, and not the QA. Perfecto is not involved because the developer user credentials are necessary and may not available per company policy.

Alternatively, we would need customers to get developer provisioning signature access, to be able to build the app with the correct signature as per our R&D documentation.

Updated 07-13-2018