FAQ

AppConnect FAQ for MobileIron Technology Partners

AppConnect Overview

What is AppConnect?

AppConnect is a MobileIron product that secures and protects enterprise mobile apps. It manages the complete lifecycle of mobile apps and app data by

- Enabling security and management features
- Distributing apps to authorized devices
- Delivering app configurations and policies to apps at runtime
- Reporting analytical usage data for apps
- Revoking app privileges as necessary

On a mobile device, AppConnect-enabled apps reside and run within a passcode protected secure container. MobileIron provides two different mechanisms to convert a mobile app into a secure app: the **AppConnect SDK** and the **AppConnect Wrapper**.

Which mobile platforms does it support? Which one should we select, the SDK or Wrapper?

Currently, the following options are available:

- iOS: **AppConnect SDK**, **AppConnect Wrapper**
- Android: **AppConnect Wrapper**

For a public iOS app, the AppConnect SDK is the only applicable option. The AppConnect Wrapper technology does not meet Apple's terms and conditions for distribution via the Apple App Store. The AppConnect Wrapper for iOS may only be used for apps that are distributed using an in-house app distribution mechanism. The Android Wrapper can be used for both public and in-house apps.

For our technology partners, we suggest you use the **AppConnect SDK for iOS** and the **AppConnect Wrapper for Android** (and this FAQ only covers these two options. That being said, if you are interested in quickly prototyping AppConnect functionality in your app, MobileIron can also offer you the AppConnect Wrapper for iOS).

MAKING ENTERPRISE MOBILE FIRST

MobileIron
415 East Middlefield Road
Mountain View, CA 94043 USA
Tel. +1.650.919.8100
Fax +1.650.919.8006
info@mobileiron.com

How does it work? What MobileIron products are required to run AppConnect apps?

Once a mobile app gets secured as an AppConnect app via the SDK or Wrapper process, it is managed by the MobileIron Virtual Smartphone platform (VSP). On the device side, the MobileIron Client, called the Mobile@Work app, should be installed and run along with the AppConnect app. If the AppConnect app needs secure tunneling to an enterprise server (AppTunnel), a MobileIron Sentry can be deployed.

What is AppTunnel? How does it work?

AppTunnel provides a secure tunneled communication path between the mobile app and the enterprise server in the corporate environment, typically running behind a firewall. The communication channels are secured by a MobileIron Sentry, which authorizes and reverse-proxies app traffic to and from the AppConnect app.

A tunneled connection is established and managed for each app. As opposed to VPN, which allows everything on the device to connect to everything inside the enterprise, AppTunnel applies app tunneling rules individually to each app on a device based on the service URLs required from the enterprise environment.

AppConnect SDK for iOS

What does the AppConnect SDK for iOS package include?

The package includes an AppConnect library, an AppConnect header file, sample programs, and a developer's manual. The current version of the AppConnect SDK for iOS supports:

- App configuration settings
- App authorization
- AppConnect passcode policy
- App tunneling
- Data Loss Prevention (DLP) Controls – open in, copy & paste, print

How much engineering effort would be needed to implement the AppConnect SDK in an iOS app?

The AppConnect SDK is fairly straightforward. You should include the AppConnect library, and at minimum, need to instantiate an AppConnect object and provide a delegate object to receive notifications from the library. MobileIron expects it would take less than a day to complete the basic integration, though more days might be needed to fully realize the value of all of the features of your app. Please see the AppConnect for iOS app developer's guide for specifics, available on the AppConnect Developer Portal.

What Data Loss Prevention (DLP) policies are supported? How does it work?

For each secured app, the following DLP options are definable in VSP:

- Open in: Allowed, Disallowed, Only to whitelisted apps, Only to AppConnect apps
- Pasteboard: Allowed, Disallowed, Only to AppConnect apps (Roadmap)
- Print: Allowed, Disallowed

Whitelisted apps are defined by their Bundle Ids. In order to make DLP work, your app is required to enforce these DLP settings as they are passed from the VSP via the Mobile@Work app. The precise UI of your app might differ from other apps depending on your app implementation.

Does the AppConnect SDK for iOS prevent Screen Capture?

No. On iOS it is not possible to prevent screen capture on a per-app basis. This is only supported on Android

Does AppConnect encrypt data in the device?

All secured apps are protected by the AppConnect passcode. Currently no data encryption is implemented in the AppConnect SDK, and we expect each app developer to use the data protection framework provided by iOS. MobileIron plans to add an additional data encryption layer in the next release of AppConnect.

Which protocols does AppTunnel support?

In the current version of AppTunnel, app connections over HTTP and HTTPS protocols are supported. Your app does not need to do anything special to take advantage of AppTunnel.

AppConnect Wrapper for Android

What benefits does the Android Wrapper provide?

Security is a big concern for all enterprises customers considering Android devices. The Android platform is an open platform and the Google Play store is not well monitored for malware. The AppConnect Wrapper for Android gives the IT administrator peace of mind by making sure that the wrapped apps

- Are password protected
- Automatically encrypt their data
- Are distributed only to authorized devices and can be disabled if required
- Are isolated from non-AppConnect applications installed by the user

The AppConnect Wrapper automatically injects these functionalities into your mobile app with no involvement from your app developer.

How is my app wrapped?

The AppConnect Wrapper converts an Android app binary (APK) to a secured AppConnect app through the following process:

1. The APK file is read to create an in-memory representation
2. Key OS level interactions are replaced by secure APIs
3. It is then repackaged to create a new binary image, the secured app
4. The secured app is signed by MobileIron

Once the app is wrapped, it can only interact with other AppConnect apps residing in the container. The app is isolated from non-AppConnect apps residing on the same device.

You can start this process by submitting your Android app APK file to MobileIron to wrap.

How are wrapped Android apps installed on an end-user's device?

Android AppConnect apps are distributed via the enterprise app store. This gives an administrator the ability to test the apps in their environment before distributing to all the end users.

The Mobile@Work app on the device is notified when the administrator chooses to make apps available. It includes a wizard UI to guide the end user through app download and install process.

Will wrapped apps on Android be available in the Google Play store?

At this time, wrapped apps will only be available through MobileIron. This has to do with preserving the security features pertinent to the Android solution (I thought the Developer could distribute directly to customer as well)

Business and Operational Process

Where should I start? How can I learn more details?

There are 3 steps to get started:

1) Read and accept the MobileIron NDA terms located at (Link)
2) Register at MobileIron University and complete the short required training
3) Contact the AppConnect Developer team once you have completed the above 2 steps to access the SDK and other tools on the AppConnect Developer Portal.

How is an AppConnect app validated to work?

When an enterprise app is secured by the AppConnect SDK or Wrapper, it should undergo rigorous testing by the app developer to validate all AppConnect functionality. MobileIron will assist in testing by providing the required system setup and a set of standard test cases. o.

Before we start integrating the AppConnect SDK into our iOS app, can we try the iOS Wrapper as a proof of concept?

Please contact MobileIron to submit a binary file that can be wrapped by MobileIron. Once complete, MobileIron will return the wrapped app for your testing and deployment. Please bear in mind that this wrapped app cannot be distributed to the public through the Apple App Store.

How is your AppConnect SDK for iOS upgraded after we include it in our app? What procedures should we go thru to ensure the functionality?

The intent is to maintain backwards compatibility with older versions of the AppConnect SDK. New releases of the VSP, Sentry, or the Mobile@Work app will not require app developers to upgrade the app.

We also intend to only add to the SDK API, rather than change it or subtract from it. If an app developer decides to upgrade to a newer AppConnect SDK version to take advantage of new features or policies, we anticipate only minimal code changes will be necessary due to the upgrade.

When our Android app is upgraded, what procedures should we go thru to get it wrapped again and ensure the AppConnect functionality?

We intend to maintain backwards compatibility with older versions of the AppConnect Wrapper. New releases of the VSP, Sentry, or the Mobile@Work app will not require app developers to upgrade the app.

When your new app release reaches the beta stage, please work with MobileIron to wrap it with the latest version for testing. We expect that app wrapping will have minimal impact from revision to revision. However, if major functionality has been added (e.g., addition of a new module to support a new file format), then it is best to re-test the wrapped app.