

Adding the MobileFirst Platform Foundation SDK to Cordova Applications

Overview

The MobileFirst Platform Foundation SDK provides a set of API methods enabling a developer to implement various MobileFirst features, such as: authentication and security mechanisms, notifications, resource requests, collecting analytics data and more.

For a complete list of MobileFirst SDK abilities visit the user documentation (http://www-01.ibm.com/support/knowledgecenter/SSHS8R_8.0.0/wl_welcome.html).

In this tutorial you will learn how to add the MobileFirst SDK to either a new or existing Cordova application created with Apache Cordova, Ionic or other third-party tool. The SDK is provided as a Cordova plugin (`cordova-plugin-mfp` (<https://www.npmjs.com/package/cordova-plugin-mfp>)). You will also learn how to configure the MobileFirst Server to recognize the application, as well as find information about the MobileFirst configuration files that are changed in the project.

The following Cordova plug-ins are available:

- **cordova-plugin-mfp**
- **cordova-plugin-mfp-push**
- **cordova-plugin-mfp-jsonstore**
- **cordova-plugin-mfp-fips**

The **cordova-plugin-mfp** plug-in contains the core MobileFirst functions and is required. If you install either the **cordova-plugin-mfp-push** plug-in, the **cordova-plugin-mfp-jsonstore** plug-in or the **cordova-plugin-mfp-fips** plug-in, the **cordova-plugin-mfp** is automatically installed.

The **cordova-plugin-mfp-jsonstore** plug-in enables your app to use JSONstore. For more information on JSONstore, see the JSONStore tutorial ([../client-side-development/jsonstore/](#)). The **cordova-plugin-mfp-push** plug-in provides permissions needed to use push notification from the MobileFirst Server for Android apps. Additional setup for using push notification is required. For more information on push notification, see the Push notification. tutorial ([../notifications/push-notifications-overview/](#)) The **cordova-plugin-mfp-fips** enable FIPS related features. For more information about FIPS, see FIPS ([../link-needs-to-be-updated](#))
TODO: Update link

Pre-requisites: MobileFirst CLI and Apache Cordova CLI installed on the developer workstation.

Make sure you have read the Setting up your MobileFirst development environment ([../setting-up-the-mobilefirst-development-environment](#)) tutorial.

Jump to:

- Adding the MobileFirst Cordova Plugin
- Registering the Cordova app in MobileFirst Server
- Running the application on emulator or on a real device
- Generated MobileFirst Native SDK artifacts
- Tutorials to follow next

Adding the MobileFirst Cordova Plugin

Before starting, make sure the MobileFirst Server is running.

From **Terminal** run the command:

```
mdo server start
```

Follow the below instructions to add the MobileFirst Cordova Plugin to either a new or existing Cordova project:

1. Create a Cordova project or use an existing one. The MobileFirst template app can be used to create a new Cordova project that already contains the MobileFirst Plugin. To use the template app, open **Terminal** and run the command:

```
cordova create myapp com.ibm.app myapp --template cordova-template-mfp
```

The **myapp** folder will be created with a copy of the cordova-template-mfp project

1. Navigate to the root of the Cordova project.
2. Add the MobileFirst Cordova Plugin(cordova-plugin-mfp (<https://www.npmjs.com/package/cordova-plugin-mfp>)) using the cordova CLI. This will install the plugin from NPM (<https://www.npmjs.com/package/cordova-plugin-mfp>):

```
cordova plugin add cordova-plugin-mfp
```

1. The optional plug-ins **cordova-plugin-mfp-push** and **cordova-plugin-mfp-jsonstore** can be added with the following commands:

- add **cordova-plugin-mfp-push**

```
cordova plugin add cordova-plugin-mfp-push
```

- add **cordova-plugin-mfp-jsonstore**

```
cordova plugin add cordova-plugin-mfp-jsonstore
```

- add **cordova-plugin-mfp-fips**

```
cordova plugin add cordova-plugin-mfp-fips
```

1. Add one or more platforms to the project using the Cordova CLI To add ios platform:

```
cordova platform add ios
```

To add android platform:

```
cordova platform add android
```

Registering the Cordova app in MobileFirst Server

1. Register the application with MobileFirst Server with the command:

```
mdo app register
```

The `mdo app register` CLI command first connects to the MobileFirst Server to register the application, followed by generating the `config.xml` file at the root of the Cordova project, and adding to it the metadata that identifies the MobileFirst Server. Each platform is registered as a native app in MobileFirst Server.

TODO: Not sure if the following is valid for Cordova

> The application registration can also be done from the MobileFirst Operations Console:

1. Open your browser of choice and load the MobileFirst Operations Console using the address `http://localhost:10080/mfpconsole/`. You can also open the console from **Terminal** using the CLI command `mdo server console`.

2. Click on the "Create new" button next to "Applications" to create a new application. Follow the on-screen instructions.

3. After successfully registering your application you can optionally download a "skeleton" Android Studio project pre-bundled with the MobileFirst Native SDK.

> **Tip:** Learn more about the various CLI commands in the [Using CLI to manage MobileFirst artifacts](../client-side-development/using-cli-to-manage-mobilefirst-artifacts/) tutorial.

Preview, emulate or run application on a device

Preview the Application

1. After registered in a MobileFirst Server, the app can be previewed with the command

```
mdo app preview
```

The preview can be done in two ways, with Simple Browser Rendering or with Mobile Browser Simulator. With Simple Browser Rendering the app is presented as a web page in your browser while Mobile Browser Simulator will simulate a Mobile Device in your browser and the app is presented inside the simulated device.

? Select how to preview your app: (Use arrow keys)

> browser: Simple browser rendering

mbs: Mobile Browser Simulator

Running the application on emulator or on a real device

1. Use the Cordova CLI to run the application on a emulator or on a real device.

To emulate the application execute the command `cordova emulate`

example for ios:

```
cordova emulate ios
```

example for android:

```
cordova emulate android
```

To run the application on a real device attached to the development machin, execute the command

cordova run example for ios:

```
cordova run ios
```

example for android:

```
cordova run android
```

MobileFirst settings in config.xml

When the MobileFirst Cordova plugin is added to the project,the file `config.xml` receives a set of new settings identified with the namespace `mfp:`. The added elements will contain information related MobileFirst features and MobileFirst Server. Here is an example of MobileFirst settings added to config.xml:

```
<mfp:android>
  <mfp:sdkChecksum>3563350808</mfp:sdkChecksum>
  <mfp:appChecksum>0</mfp:appChecksum>
  <mfp:security>
    <mfp:encryptWebResources enabled="false" />
    <mfp:testWebResourcesChecksum enabled="false" ignoreFileExtensions="png
, jpg, jpeg, gif, mp4, mp3" />
  </mfp:security>
</mfp:android>
<mfp:platformVersion>8.0.0.00-20151214-1255</mfp:platformVersion>
<mfp:clientCustomInit enabled="false" />
<mfp:server runtime="mfp" url="http://10.0.0.1:9080" />
<mfp:directUpdateAuthenticityPublicKey />
<mfp:languagePreferences>en</mfp:languagePreferences>
```

- **mfp:android:** Root element for settings specific for the android platform
- **mfp:ios:** Root element for settings specific for the ios platform
- **mfp:windows:** Root element for settings specific for the windows platform
- **mfp:sdkChecksum:** Checksum of the SDK in use
- **mfp:appChecksum:** Checksum of the app
- **mfp:security:** Root element for security configurations
- **mfp:encryptWebResources:** ??
- **mfp:testWebResourcesChecksum:** Enables or Disables the test for web resources checksum
- **mfp:platformVersion:** ??
- **mfp:clientCustomInit:** ??
- **mfp:server:** MobileFirst Server URL and Runtime definition
- **mfp:directUpdateAuthenticityPublicKey:** MobileFirst Server URL and Runtime definition
- **mfp:languagePreferences:** Default language for client sdk system messages (en, fr, es)

Editing MobileFirst settings in config.xml with MobileFirst CLI

The MobileFirst CLI can be used to edit the MobileFirst settings in config.xml with the command

```
mdo app config
```

Tutorials to follow next

Now that the MobileFirst Cordova plugin is added to the application you can continue reading tutorials for Cordova development ([../hybrid-tutorials/](#)) to learn more about authentication and security, server-side development, notifications, and more.