

Authenticity in Native iOS applications

This is a continuation of the Application Authenticity Protection (../) tutorial.

application-descriptor.xml

Adding the security test

Modify the `application-descriptor.xml` file of your application.

Add the `securityTest` attribute to the Android or iPhone/iPad environment element. For example:

```
<iphone bundleId="com.worklight.MyBankApp" applicationId="MyBankApp" securityTest="customTests" version="1.0">
```

Specifying the bundleId and applicationId

1. Specify the `bundleId` of your application exactly as you defined it in the **Apple Developer** portal.



It can be added either in the Source view:

```
<iphone bundleId="com.worklight.MyBankApp" version="1.0">
```

Or in the Application Description Editor (design view):

application-descriptor.xml

Native iOS Application Descriptor

Details

Id*:
The ID must be an alphanumeric string that starts with a letter. It may contain underscore ("_") characters. The ID must not be a reserved word in JavaScript. Once the application id has been defined it cannot be modified.

Display name*:
Application name. Appears in the MobileFirst Console and is copied to the descriptor files of various web and desktop environments

Description*:

Appears in the MobileFirst Console.

Version*:

Security test:
Specifies a security configuration defined in authenticationConfig.xml. When a client attempts to access a protected resource, MobileFirst checks whether the client is already authenticated according to the security test. If the client is not yet authenticated, MobileFirst runtime triggers the process of obtaining the client credentials and verifying them.

Bundle id:
 Application Id:

To enable an application authenticity check, you must specify the application id. This value must match the value of the application id property in the worklight.plist file.

Access token expiration (seconds):
Expiration period of OAuth access token. The default is 3600 seconds (one hour).

User identity realms:
Comma separated ordered list of user identity realms for OAuth authentication. The order dictates the selected user identity. If the list is empty, the ID token contains no identity information.

▼ pushSender (optional)

password*:

- Specify the `applicationId`. The Application Id value must match the value of the application id property, which is located in the `worklight.plist` file.

It can be added either in the Source view:

```
<iphone bundleId="com.worklight.MyBankApp" applicationId="MyBankApp" securityTest="custom Tests" version="1.0">
```

Or in the Application Description Editor (design view):

application-descriptor.xml

Native iOS Application Descriptor

Details

Id*:

The ID must be an alphanumeric string that starts with a letter. It may contain underscore (" _ ") characters. The ID must not be a reserved word in JavaScript. Once the application id has been defined it cannot be modified.

Display name*:

Application name. Appears in the MobileFirst Console and is copied to the descriptor files of various web and desktop environments

Description*:

Appears in the MobileFirst Console.

Version*:

Security test:

Specifies a security configuration defined in authenticationConfig.xml. When a client attempts to access a protected resource, MobileFirst checks whether the client is already authenticated according to the security test. If the client is not yet authenticated, MobileFirst runtime triggers the process of obtaining the client credentials and verifying them.

Bundle id:

Application Id:

To enable an application authenticity check, you must specify the application id. This value must match the value of the application id property in the worklight.plist file.

Access token expiration (seconds):

Expiration period of OAuth access token. The default is 3600 seconds (one hour).

User identity realms:

Comma separated ordered list of user identity realms for OAuth authentication. The order dictates the selected user identity. If the list is empty, the ID token contains no identity information.

▼ pushSender (optional)

password*:

3. In Xcode, verify that the following value exists in the **Other Linker Flags** field: *-ObjC*