Background Authorize

BlackBerry Dynamics Application Developer Guide

Contents

Introduction

Background Authorize is a feature for BlackBerry Dynamics applications. It enables a recently locked application to utilize the principal BlackBerry Dynamics APIs like secure storage and secure communications when the application is running in the background. With this capability an application can unlock the container without a user's intervention.

This feature is helpful when an application may have stopped (because the operating system unloaded it from runtime memory, or the application has crashed). On an iOS device an application may be started in the background in response to having received an APNS message (for example, a new email has been received). In this scenario, if the Background Authorize feature is enabled, the application is able to download the new messages and store them in the secure container. When a user brings the application to the foreground, they are required to authorize and can access their new messages.

Use of the Background Authorize feature is an application privilege. It must be requested from BlackBerry. Your application must implement the feature according to the rules and standards detailed in this document.

Availability

The Background Authorize feature is supported from BlackBerry Dynamics SDK for iOS version 6.0 and BlackBerry Dynamics SDK for Android version 9.1.

Background Authorize in the BlackBerry Dynamics authentication cycle

A BlackBerry Dynamics application goes through the following authentication states at start-up.

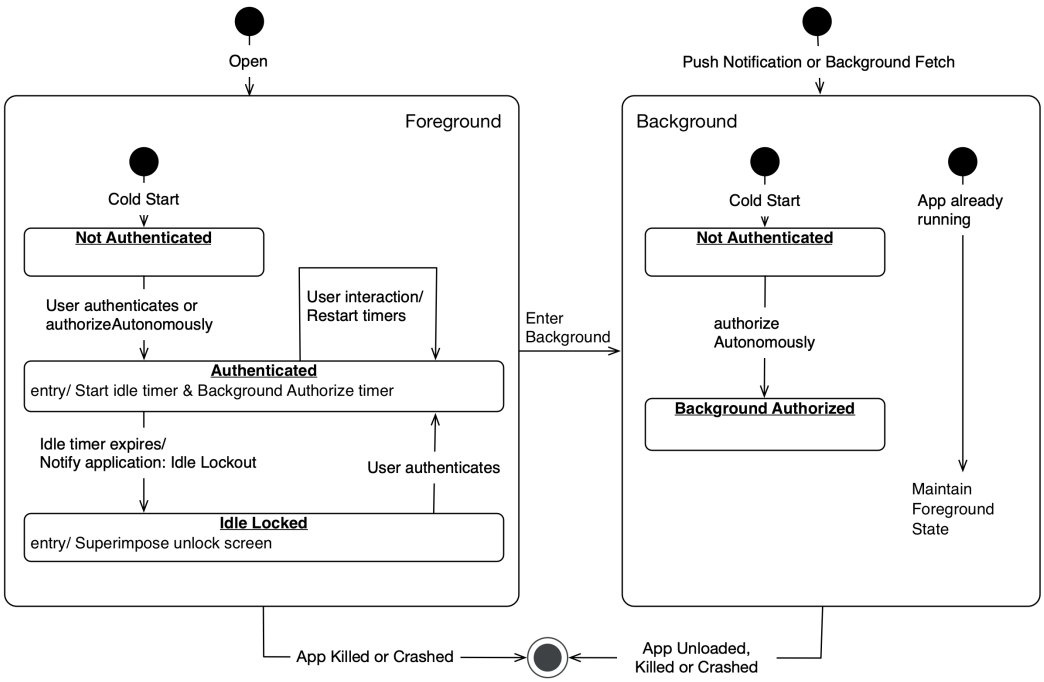
- Not authenticated: The initial state. The application will be in this state until the end user has authenticated for the first time after start-up. Note that the end user could supply a different authentication secret, instead of a password, if a Trusted Authenticator is in use or no password.
- Authenticated: The state immediately after the user has supplied their password, or means of authentication. Data on the device is protected with an encryption key that is derived from the secret. After the secret has been supplied, the key is derived and the runtime can access its management data, and any application data.
- Idle: The state entered when the user hasn't interacted with the application for the idle time out. The duration of the idle time out is set in the management console, as a security policy setting. The runtime locks the application user interface when the Idle state is entered.

Any of the above states can apply whether the application is running in the foreground or in the background.

- Background Authorized: The state entered when the application has been started in the background as a result of receiving a push notification or background fetch operation and the conditions for autonomous authentication have been met.

Diagram

The above states are illustrated in the following diagram.



Name	st08 Background Authorize
Documentation	UML State Transition diagram for Background Authorize

If the device is switched off and on, and then the application is started again, it will be back in the initial state: Not authenticated. The same is true when the application is started after having been unloaded from memory. The application could be unloaded by the device operating system to release resources, or if it crashes, or if the user chooses to terminate it.

Background Authorize Policy

Background Authorize can be allowed or disallowed by enterprise policy. This is implemented using the BlackBerry Dynamics Application Policies feature which is supported by any Good Control (GC) or Unified Endpoint Management (UEM) Server.

The end user of a BlackBerry Dynamics application is always associated with an enterprise, by activation. The enterprise adminstrator can configure policies and profiles that control use of the application by the end user, including allowing or disallowing Background Authorize.

Background Authorize should be disabled by default and only enabled by an enterprise administrator updating the configuration.

Background Authorize Registration

Access to Background Authorize is restricted and must be requested from BlackBerry. Applications that are granted access to Background Authorize are registered by BlackBerry, and each issued a unique registration message. The message must be embedded in the application declaration, at build-time.

Summary of Background Authorize conditions

The conditions for the application to enter the Background Authorize state are:

- The application has started in the background.
- The application attempts to authorizeAutonomously.
- Background Authorize is allowed by enterprise policy.
- App has been unlocked at least once since a device restart
- The end user last authenticated within the time period defined in the background authorize policy, e.g. within the last 2 days.
- A valid Background Authorize registration message is embedded in the application.
- Device is not running in low power mode.

Implementation Essentials

To implement Background Authorize in your application, you would typically do the following:

- Create a Background Authorize policy definition for your application.
- Implement use of authorizeAutonomously (iOS) or serviceInit (Android) when application is started in the background.

Create a Background Authorize policy definition

Define a custom application policy that will control use of Background Authorize by end users.

- Create an Application Policies definition file, if your application doesn't already have one.

  For more inforamtion, see Technical Brief: BlackBerry Dynamics App Policies

- Add a setting for the Background Authorize policy.

  The following snippet shows an example setting definition.

```
<setting name="GD_SDK_Security_AllowBackgroundAuthorize">
    <select>
        <key>GD_SDK_Security_AllowBackgroundAuthorize</key>
        <label>Period:</label>
        <options ref="GD_SDK_Security_AllowBackgroundAuthorizeOptions" />
    </select>
</setting>
<dl name="GD_SDK_Security_AllowBackgroundAuthorizeOptions" dtype="number">
    <dv><desc>Off</desc><value>0</value></dv>
    <dv><desc>On: Half an hour</desc><value>30</value></dv>
    <dv><desc>On: One day</desc><value>1440</value></dv>
    <dv><desc>On: Three days</desc><value>4320</value></dv>
</dl>
```

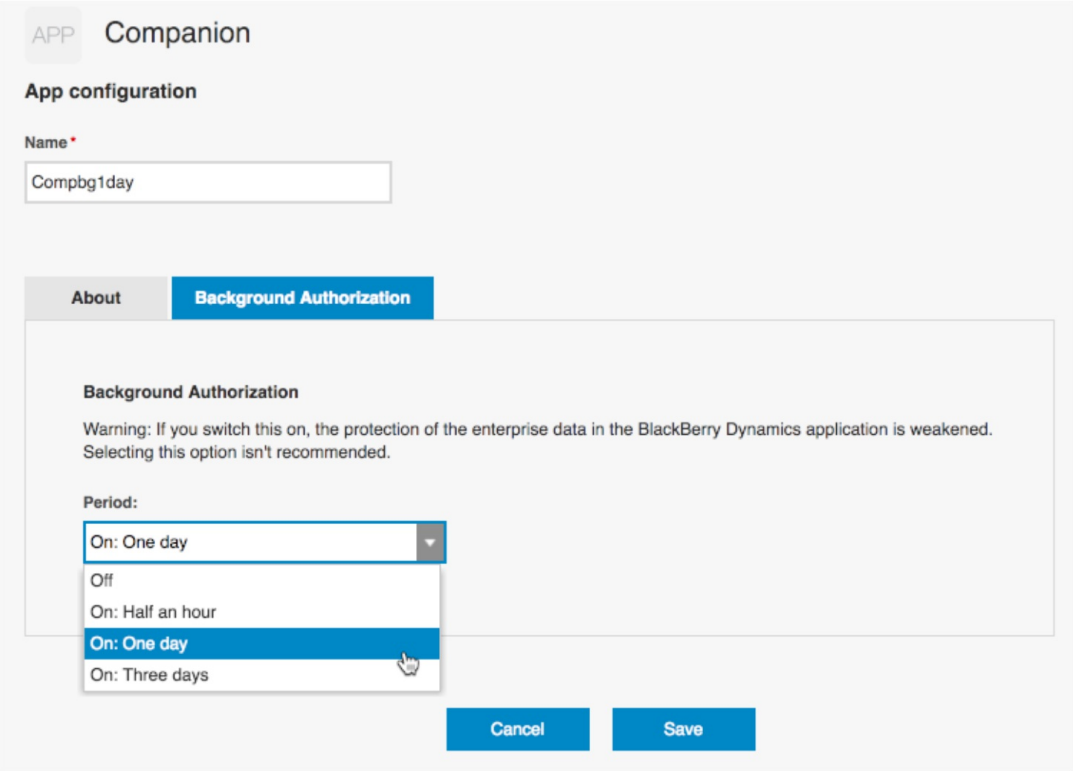
  The name of the key must be set to: GD_SDK_Security_AllowBackgroundAuthorize

- Include the setting in the structural part of the Application Policies file.

  The following snippet shows an example of the structural part of the definition.

```
<pview>
    <pview type="tabbed" key="background">
        <title>Background Authorization</title>
        <desc>Warning: If you switch this on, the protection of the enterprise
        data in the BlackBerry Dynamics application is weakened. Selecting this
        option isn't recommended.</desc>
        <pe ref="GD_SDK_Security_AllowBackgroundAuthorize" />
    </pview>
</pview>
```

- Upload the definition, or a revision, in the UEM management console. On the **BlackBerry Dynamics** tab for your application, next to **App configuration**, click **Upload a template**.



- To check your policy you can optionally call the `getApplicationPolicy` or `getApplicationPolicyString` function to get the settings as a collection or JSON string, respectively.

Implement usage of Background Authorize

Your application should call `GDiOS.authorizeAutonomously` or `GDAndroid.serviceInit` when the application is launched in background. This could happen if any of the following are in use:

- Apple Push Notification Service (APNS) or Firebase Cloud Messaging (FCM)
- Background fetch or background tasks.

Authorization processing can complete in the background, if all the conditions are met. See 'Summary of Background Authorize conditions' above.

Call this function from any callback that handles background launch, for example on iOS the application `didReceiveRemoteNotification:` or `performFetchWithCompletionHandler:` function.

Access the `canAuthorizeAutonomously` property first, to check that a background authorize or no password policy applies to the current end user and that autonomous authorization is possible.

A **GDAppEvent** notification will be sent when an application has attempted to access its data as a result of the system starting the application process from cold in the background. The following **GDAppEventTypes** enumerations could be returned.

- `GDAppEventBackgroundAuthorized` when authorized.
- `GDAppEventBackgroundNotAuthorized` when not authorized.

Check **GDAppResultCode** to determine the failure reason when not authorized.

Registration

Access to Background Authorize is restricted and must be requested from BlackBerry. Applications that are granted access to Background Authorize are registered by BlackBerry, and each issued a unique registration message. The message must be embedded in the application declaration, at build-time.

Procedure

To access the Background Authorize feature, you must register your application with BlackBerry.

1. Send a Background Authorize access request.

   Send an email message to BlackBerryDynamicsRegistrar@blackberry.com with the following information:

   - Application Package/Bundle information:
     - For iOS your applications Info.plist
     - For Android your application AndroidManifest.xml and settings.json file
   - The Application Policies definition XML file.
   - Provide details of the following:
     - Type of application.
     - What data would be updated while the application is background authorized
     - How implementing this feature would improve the user experience of your application.
     - Background Authorize options for the enterprise administrator and end user, if any.

2. Answer any questions from the registrar.

   In a reply the registrar may ask you some questions.

   The registrar may in some cases request changes to your application or to the application policy.

   The registrar may initially only grant access to this feature for a fixed length beta period using a test application identifier.

3. Build the application with modified application declarations from the registrar.

   When all changes have been made, the registrar will issue a *Background Authorize registration message*. The form of the message will be some signed data that you insert into your application's Info.plist file. See below for an example registration message.

Example Registration Messages

The following is a sample Background Authorize registration message for iOS. It would be inserted into the application Info.plist file. The signature has been shortened for convenience.

```
<key>GDPermissions</key>
<array>
    <dict>
        <key>Permission</key>
        <dict>
            <key>BackgroundAuthorizePermission</key>
            <string>1</string>
            <key>GDApplicationId</key>
            <string>com.company.example</string>
            <key>nativeApplicationId</key>
            <string>com.company.example</string>
        </dict>
        <key>Signature</key>
        <string>0b9aeff...ced54863754d9</string>
        <key>SignatureScheme</key>
        <string>RSAv1</string>
    </dict>
</array>
```

When to make a subsequent registration request

The registration message generally remains valid through updates to your application and through new versions of the SDK. However, you will need to request access again if:

- The bundle identifier/package name changes.
- The entitlement identifier (generally known as the GD Application Identifier, or GD App ID) changes.
- There will be a substantial change in the frequency or type of data transferred while application is running in the background.

Legal Notice

Legal Information