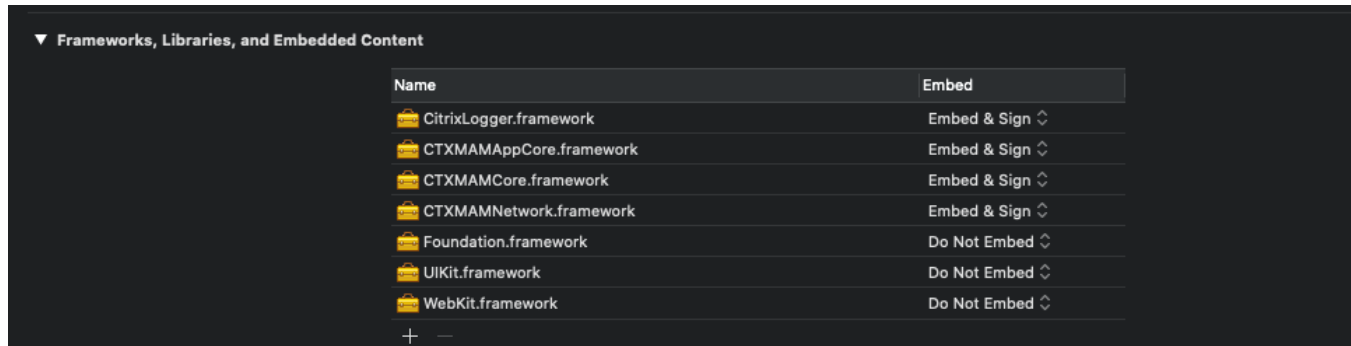


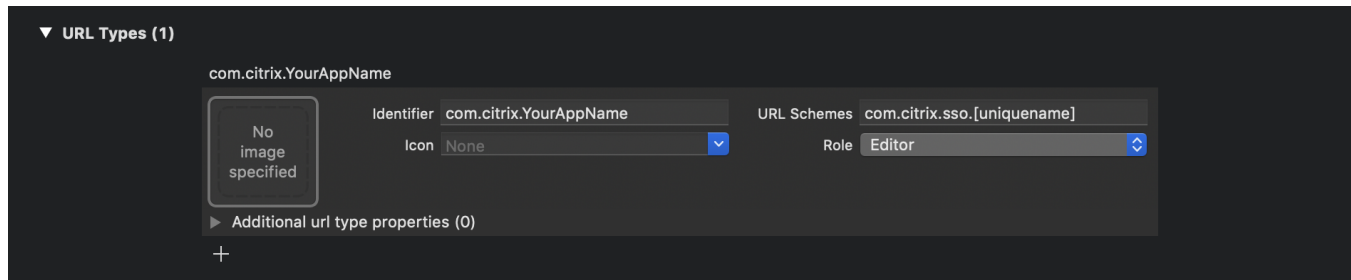
Citrix MAM iOS SDKs - Network SDK - Developer Guide

1. Add CTXMAMCore.framework, CTXMAMAppCore.framework, CTXMAMNetwork.framework, and CitrixLogger.framework to your project. On the project properties under the "General" tab add all 3 frameworks listed above to the "Embedded Binaries" section.

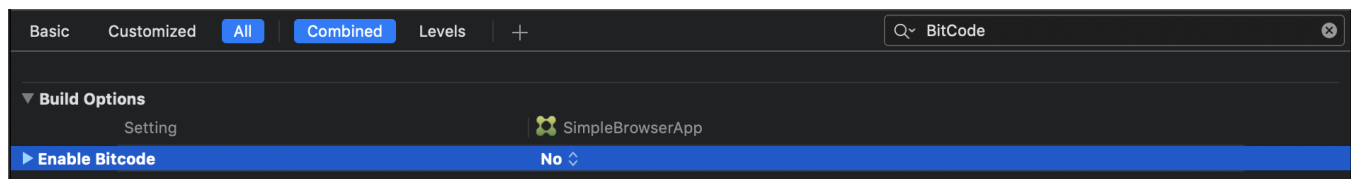


2. On the project target properties under the "Info" tab, add a new URL Type with URL Scheme which starts with com.citrix.sso.[PACKAGEID], if an URL Scheme with "com.citrix.sso." prefix was not previously added. Ensure that you run uuidgen command in your command line and paste the newly generated UUID in the PACKAGEID field. (You can generate a package ID for your application by running uuidgen at the terminal app)

Keep in mind that this same package ID will also be used in subsequent steps below.

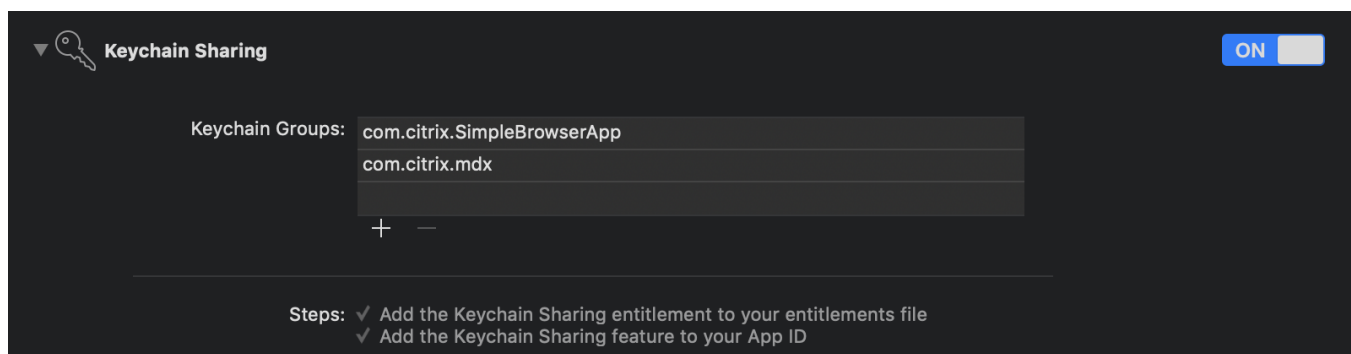


3. On the project properties under the "Build Settings" tab set "Enable BitCode" to "No"



4. On the project properties under "Capabilities" tab set "Keychain Sharing" to "On"

4A. Add com.citrix.mdx to the list of "Keychain Groups:"



5. Import <CTXMAMCore/CTXMAMCore.h> into your AppDelegate.m file.

Import headers code block

```
//
// AppDelegate.m
// Your App Name here
//
// Created by Your Name Here on 3/21/19.
// Copyright © 2019 Your Company Name Here, LLC. All rights reserved.
//

#import "AppDelegate.h"
#import <CTXMAMCore/CTXMAMCore.h>
```

6. In the AppDelegate.m in your application didFinishLaunchingWithOptions add a call to initialize the SDKs, [CTXMAMCore initializeSDKs].

Application Did Finish Launching override

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Override point for customization after application launch.
    [CTXMAMCore initializeSDKs];
    return YES;
}
```

7. Back on the project properties under the "Build Phases" tab add a new "Run Script" phase with the following script.

MDX File Creation Script

```
# Type a script or drag a script file from your workspace to insert its path.
export STOREURL="http://yourstore.yourdomain.com"
export APPTYPE="sdkapp"
export PACKAGEID="" #Please run uuidgen at the command line and paste the output value in the PACKAGEID
variable, This has be the same UUID which was generated in step 2.
export APPIDPREFIX=""#Your apps prefix id which can be found in your apple's developer account page.
export TOOLKIT_DIR="$PROJECT_DIR/../Tools"

if [ -z "${PACKAGEID}" ]
then
    echo "PACKAGEID variable was not found or was empty, please run uuidgen at the command line and paste the
output value in the PACKAGEID variable in your post build script."
    exit 1
fi

if [ -z "${APPIDPREFIX}" ]
then
    echo "APPIDPREFIX variable was not found or was empty, please refer to the \"how to\" document located in
the documentation folder of the SDK package on where to find your Apple's application prefix ID."
    exit 1
fi
if [ ! -d $TOOLKIT_DIR/logs ]
then
    mkdir $TOOLKIT_DIR/logs
fi

"$TOOLKIT_DIR/CGAppCLPrepTool" SdkPrep -in "$CONFIGURATION_BUILD_DIR/$EXECUTABLE_FOLDER_PATH" -out
"$CONFIGURATION_BUILD_DIR/$EXECUTABLE_NAME.mdx" -storeURL "${STOREURL}" -appType "${APPTYPE}" -packageId
"${PACKAGEID}" -entitlements "$SRCROOT/$PROJECT/$PROJECT.entitlements" -appIdPrefix "${APPIDPREFIX}" -
minPlatform "9.0"
```

7a. Ensure that you run `uuidgen` command in your command line and paste the newly generated uuid in the PACKAGEID field. This has be the same UUID which was generated in step 2.

7b. Ensure that you put your app id prefix in the APPIDPREFIX field. This ID can be found in the Apple Developer portal under the AppIDs section. This is usually the team ID but it can be different.

Edit your App ID Configuration

RemoveSave

Platform	App ID Prefix
iOS, tvOS, watchOS	Here is the ID (Team ID)
Description	Bundle ID
<input type="text" value="Citrix Secure Hub Test ID"/>	com.citrix.mdxhub (explicit)

You cannot use special characters such as @, &, *, ' , "

7c. Ensure that the TOOLKIT_DIR property points to the Tools folder in the SDK package.

Embedding your IPA file into the MDX file for publication on the CEM Server

1. Create you IPA by going thru the archiving process in Xcode.
2. Run the following command:

Embedding IPA File

```
export TOOLKIT_DIR="$PROJECT_DIR/../Tools"
export IPA_FILE_PATH="Provide IPA File Path"
export EXECUTABLE_NAME="Provide the name for your application's app folder."
#CONFIGURATION_BUILD_DIR comes from Xcode.

"$TOOLKIT_DIR/CGAppCLPrepTool" SetInfo -in "$CONFIGURATION_BUILD_DIR/$EXECUTABLE_NAME.mdx" -out
"$CONFIGURATION_BUILD_DIR/$EXECUTABLE_NAME.mdx" -embedBundle "${IPA_FILE_PATH}"
```

Extra Notes

You are required to have icons set up for your app. In particular, your 60x60 icon (the 20x20@3x icon) must be in PNG format with the name "AppIcon60x60.png"