Building AIR Apps for iOS

January 22, 2013

Link: <http://www.adobe.com/devnet/air/articles/packaging-air-apps-ios.html>

# Installing OpenSSL for Windows:

1. Download and run the Open SSL installer (<http://slproweb.com/download/Win32OpenSSL_Light-1_0_1c.exe>)
2. When the installer prompts for installation location – Windows or /bin – select the /bin option
3. Complete installation

# Generating new iOS Developer Certificates

1. If the previously generated certificate “iOS\_Dev.cer” still works, skip to “Configuring Flash Builder to Use iOS Developer Certificates.”
2. Log into the Apple Developer Center (<http://developer.apple.com>) or create an account
3. Navigate to the iOS Provisioning Portal and select the Certificates tab in the left pane
4. Revoke any listed development certificate(s)
5. If this is a new application, create a new App ID under the App IDs tab in the left pane
   1. The Bundle Identifier should have the same value as the <id> tag in your Application.xml file
6. Under the Devices (also left pane) interface, add a new device, again, if necessary
   1. The UUID of the device can be found by connecting to iTunes
7. Run Command Prompt as Administrator
8. To generate a Certificate Signing Request, enter the following commands:
   1. cd C:\OpenSSL-Win32\bin
   2. openssl
   3. genrsa –out mykey.key 2048
   4. req –new key mykey.key –out CertificateSigningRequest.certSigningRequest –subj “/emailAddress=<accountEmailAddress>, CN=<accountName>, C=US”
      1. **Note:** replace <accountEmailAddress> and <accountName> with the developer account’s email address and developer name, respectively
   5. Navigate to the Certificates interface on the Apple developer website and, under the Development tab, select Request Certificate
   6. Browse to C:\OpenSSL-Win32\bin and select CertificateSigningRequest.certSigningRequest
   7. After a short while, refresh the page and download (or download then move) the generated certificate to the OpenSSL\bin directory
   8. Download and install the Apple WWDR certificate (<http://developer.apple.com/certificationauthority/AppleWWDRCA.cer>)
   9. In the same Command Prompt, enter the following commands:
      1. x509 –in ios\_development.cer –inform DER –out ios\_development.pem –outform PEM
      2. pkcs12 –export –inkey mykey.key –in ios\_development.pem –out C:\%userprofile\iOS\_Dev.p12
      3. Enter a password for the certificate. This is password that will be needed when testing the application.
9. Return to the iOS Provisioning Portal and select the Provisioning tab from the left pane
10. Fill out the required information, save, and download the resulting .mobileprovision.

# Configuring Flash Builder to Use iOS Developer Certificates

1. Set the Target platform to Apple iOS
2. Set the Launch method to On device and the Build Target to device
3. Click on the ‘Configure packaging settings’ link
4. For the Certificate input, browse to and select the iOS\_Dev.p12 file (either located in your home directory or in Dropbox/GitHub if it exists)
5. For the Provisioning file input, browse and select the .mobileprovision file downloaded previously
6. Click OK
7. Choose Debug via USB for debugging method.
   1. If the program makes a notification about requiring AIR 3.4 or higher, follow the steps in “Updating Flash Builder AIR SDK” below
8. Press the Debug button and if everything went according to plan, the application will appear on your USB connected Apple device and debugging will start

# Updating Flash Builder’s AIR SDK (because auto updates are too easy)

1. Download the latest AIR SDK (<http://www.adobe.com/devnet/air/air-sdk-download.html>)
2. Open the downloaded zip file
3. Navigate another Windows Explorer to C:\Program Files\Adobe\Adobe Flash Builder 4.7 (64 Bit)\sdks (or the equivalent)
4. Make a copy of the most current SDK
5. Open the original folder and copy into it the contents of the downloaded SDK
   1. Select ‘Copy and Replace’ and ‘Do this for the next x items’
6. Navigate to C:\Program Files\Adobe\Adobe Flash Builder 4.7 (64 Bit)\eclipse\plugins and open the folder that matches com.adobe.flash.compiler\_\*
   1. If multiples exist, choose the most recent
7. Make a copy of AIRSDK and then copy the downloaded SDK into AIRSDK, again Copying and Replacing all files
8. Edit AIRSDK\air-sdk-description.xml and change the SDK version and description to the version of the downloaded SDK
9. The SDK has now been updated. In order to use the new SDK in the application(s), edit the application.xml file and change the <application> tag at the beginning of the file to reflect the current version of the new SDK
10. Debugging through USB should now work

Final Note:

If the instructions in this guide do not work and there are none found elsewhere, stop what you are doing and switch from Flex to HTML5.