Additional Information

Working with bitcode in iOS apps

- The application-authenticity security check is not supported with bitcode.
- watchOS applications require bitcode enabled.

To enable bitcode, in your Xcode project navigate to the **Build Settings** tab and set **Enable Bitcode** to **Yes**.

Enforcing TLS-secure connections in iOS apps

Starting from iOS 9, Transport Layer Security (TLS) protocol version 1.2 must be enforced in all apps. You can disable this protocol and bypass the iOS 9 requirement for development purposes.

Apple App Transport Security (ATS) is a new feature of iOS 9 that enforces best practices for connections between the app and the server. By default, this feature enforces some connection requirements that improve security. These include client-side HTTPS requests and server-side certificates and connection ciphers that conform to Transport Layer Security (TLS) version 1.2 using forward secrecy.

For d**evelopment purposes**, you can override the default behavior by specifying an exception in the info.plist file in your app, as described in App Transport Security Technote. However, in a **full production** environment, all iOS apps must enforce TLS-secure connections for them to work properly.

To enable non-TLS connections, the following exception must appear in the **project-name-info.plist** file in the **project-name\Resources** folder:

```
<key>NSExceptionDomains</key>
  <dict>
    <key>yourserver.com</key>

      <dict>
      <!--Include to allow subdomains-->
      <key>NSIncludesSubdomains</key>
      <true/>

      <!--Include to allow insecure HTTP requests-->
      <key>NSTemporaryExceptionAllowsInsecureHTTPLoads</key>
      <true/>
    </dict>
  </dict>
```

To prepare for production

1. Remove, or comment out the code that appears earlier in this page.
2. Set up the client to send HTTPS requests by using the following entry to the dictionary:

   ```
   <key>protocol</key>
   <string>https</string>

   <key>port</key>
   <string>10443</string>
   ```

   The SSL port number is defined on the server in **server.xml** in the `httpEndpoint` definition.
3. Configure a server that is enabled for the TLS 1.2 protocol. For more information, see Configuring MobileFirst Server to enable TLS V1.2 (http://www-01.ibm.com/support/docview.wss?uid=swg21965659)

4. Make settings for ciphers and certificates, as they apply to your setup. For more information, see App Transport Security Technote (https://developer.apple.com/library/prerelease/ios/technotes/App-Transport-Security-Technote/), Secure communications using Secure Sockets Layer (SSL) for WebSphere® Application Server Network Deployment (http://www-01.ibm.com/support/knowledgecenter/SSAW57_8.5.5/com.ibm.websphere.nd.doc/ae/csec_sslsecurecom.html?cp=SSAW57_8.5.5%2F1-8-2-33-4-0&lang=en), and Enabling SSL communication for the Liberty profile (http://www-01.ibm.com/support/knowledgecenter/SSAW57_8.5.5/com.ibm.websphere.wlp.nd.doc/ae/twlp_sec_ssl.html?cp=SSAW57_8.5.5%2F1-3-11-0-4-1-0).