Nintex Mobile Enterprise Deployment Guide

Table of Contents

Nintex Mobile Enterprise Apps for Android

Android Enterprise Deployment Notes

When building an Android custom application using the Nintex Mobile Enterprise service, an Android application package (.apk) file is generated and available for download. This Android package is signed with a certificate issued by a recognised certificate authority, which allows companies deploy the unaltered package to devices.

There are no specific distribution requirements for Android apps.

Nintex Mobile Enterprise Apps for Windows Phone

Windows Phone Enterprise Deployment Notes

When building an enterprise Windows Phone application using the Nintex Mobile Enterprise service, a Windows Phone application package (.xap) file is generated and available for download.

Windows Phones enable companies to publish and distribute Windows Phone apps directly to their users, bypassing the Windows Phone Store. However, before being able to install signed enterprise apps, the users have to enroll their phones for app distribution with their companies.

To distribute Windows 8 Enterprise Phone apps

1. Register for a company developer account.
2. Acquire an enterprise certificate.
3. Create an application enrollment token (AET).
4. Prepare enterprise apps for distribution.
5. Develop a Company Hub app (optional).

Once the company has set up their distribution systems, the users can enroll their devices and install company enterprise apps through one of the following formats: a Mobile Device Management (MDM) solution, an email, or the Company Hub app.

Note: This document will not discuss the Company hub app distribution model.

Enterprise Code Signing Certificate

A Symantec Enterprise Code Signing Certificate protects users from downloading corrupted files. In order to safely distribute apps to company users, all Windows Phone enterprise applications must be signed using a Symantec Enterprise Code Signing Certificate. To obtain this certificate, each company must have a verified company account on Dev Center for Windows Phones. For more information on Windows Dev Center, see Registration Info.

Once an account is established, a company can acquire a Symantec Enterprise Mobile Code Signing Certificate. The certificate is used to generate an application enrollment token (AET) that enrolls users' devices and signs enterprise apps.

To obtain a Symantec Enterprise Mobile Code Signing Certificate

1. Retrieve the company's Symantec ID from the company's Dev Center account.
2. Visit Enterprise Mobile Code Signing Certificate on Symantec's website to complete the required steps.
3. Record which computer and browser requested the certificate and return to that particular browser to collect the certificate.

Note: Do not use private browsing mode; this can prevent the retrieval of the certificate.

4. Import the certificate into the store.
5. Go to certificates snap-ins to export the certificate in PFX format.

For more information on installing the Symantec Enterprise Mobile Code Signing Certificate, see How to install the Windows Phone Private Enterprise Root and Intermediate certificates.

For more information on exporting the certificate, see Export a Certificate with the Private Key.

Application Enrollment Token

An application enrollment token (AET) is used to enroll users' phones on company accounts and is a prerequisite for installing apps published by the companies. This process can be facilitated by a Mobile Device Management (MDM) solution. However, some MDM solutions may require the company to upload an AET to support Windows Phone deployments. Yet, if a company is deploying Windows Phone enterprise apps without an MDM solution, or if the MDM solution requires an AET, the company should use the AETGenerator tool to create an AET.

For more information on AETGenerator, see How to generate an application enrollment token for Windows Phone.

Enrolling Devices for Enterprise Distribution

Certain MDM solutions automatically generate and deploy AET files as well as enroll users' devices. If a company is not using one of the automatic MDM solutions, the company must complete the process manually.

Once an AET is ready for delivery, the company sends it to the users. The AET (.aetx) file is distributed through email or a secure website, and users can access the site from their phones. However, if a company distributes the file through email, Microsoft recommends applying Information Rights Management (IRM) protection and renaming the AET file.

Once the company sends out the file, the users tap the AET on their phones to enroll for company app distribution. Once completed, the phone is enrolled for the duration of the valid certificate. Through this process, users cannot unenroll their phones by using the phone UI. However, since Windows Phones are not restricted to a single company account, the users can enroll in multiple company accounts by installing different AET files. This allows users to not be restricted to a single AET. Once the certificate expires, it can no longer sign Windows Phone apps.

Understanding device enrollment

After enrolling for company app distribution, the AET is installed on a secure data store in the phone. Once secure, the phone sends the AET Publisher ID to a Microsoft service to confirm the company account validity.

The phone automatically attempts to validate the AET when:

- Enrolling it on their phones.
- Installing a company signed and published app.
- Starting an installed company app.
- Contacting Microsoft services for company account validity.

The validation of the AET includes a signature validation, a certificate chain validation to a specific root certificate, and a date check on the validity period of the certificate. If the AET fails to validate during any of these instances, the task associated fails.

Signing an Enterprise App

Before company apps can successfully be distributed, Windows Phone enterprise apps must be signed with the installed enterprise mobile code signing certificate. Windows Phone development tools provides command-line tools to sign Windows Phone application package (.xap) files.

To sign Windows Phone enterprise apps

1. Download Windows Phone development tools at Windows Phone SDK archives.
2. Ensure the enterprise mobile code signing certificate and secret key password are exported to a PFX file.
3. Start Visual Studio Native Tools Command Prompt and **Run as administrator**.
   ```
   c:\cd %ProgramFiles(x86)%\Microsoft SDKs\Windows
   Phone\v8.1\Tools\XapSignTool
   ```
4. Use the following command to sign the enterprise app:
   ```
   xapsigntool sign /v /signtool "C:\Program Files (x86)\Windows
   Kits\8.1\bin\x86" /f certificate.pfx /p password
   NintexMobileEnterprise.xap
   ```

   Where *certificate.pfx* is the filename of the exported enterprise mobile code signing certificate.

   Where *password* is the secret key password.

   Where *NintexMobileEnterprise.xap* is the filename of the NintexMobileEnterprise generated .xap file.

When the .xap file is signed, it is updated and ready to be deployed to enrolled devices through email or an MDM.

For more information on signing apps for distribution, see Preparing company apps for distribution for Windows Phone.

Nintex Mobile Enterprise Apps for iOS

iOS Enterprise Deployment Notes

When building an iOS custom application using the Nintex Mobile Enterprise service, an iOS application package (.ipa) file is generated and available for download. This iOS build is signed by the Nintex development distribution certificate when created. However, the build cannot be distributed until it is signed by the company's enterprise distribution certificate, which overwrites the Nintex signature.

Apple Developer Enterprise Program

To distribute an iOS Enterprise App, a company must be registered in the iOS Developer Enterprise Program. Enrollment is available at iOS Developer Enterprise Program.

Enterprise Distribution Certificate

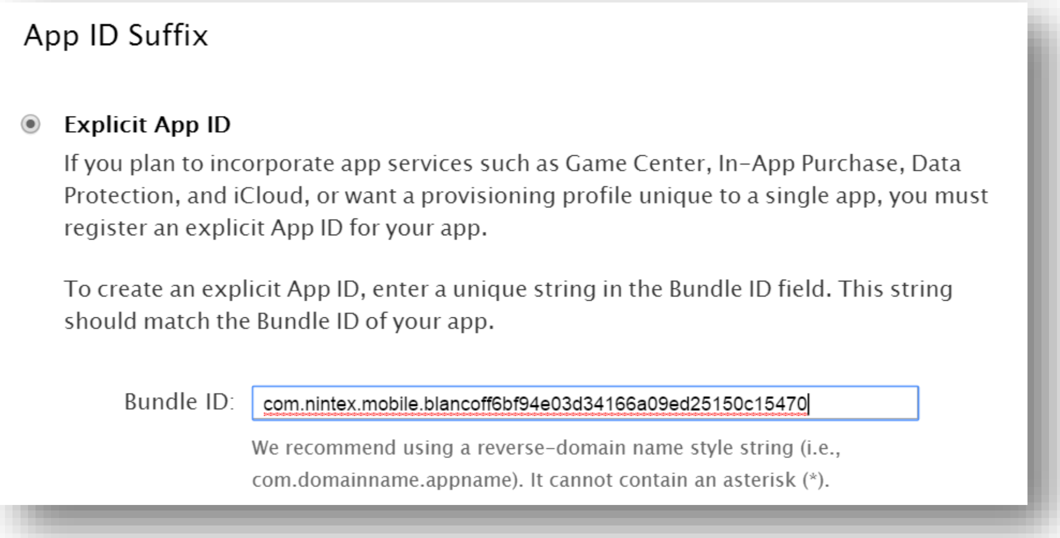
Once registered in the iOS Developer Enterprise Program, an organisation can generate an Enterprise Distribution Certificate through the Apple Developer Portal.

To distribute iOS Enterprise apps

1. Register an App ID
2. Create a distribution certificate
3. Generate a Mobile Provisioning Profile
4. Distribute an enterprise apps

Register an App ID (aka Bundle ID)

1. When a Nintex Mobile Enterprise application has been defined, it is automatically given an App ID. The ID is visible under the **Deployment** tab. You must register an App ID within the Apple Developer Portal that matches this App ID value.
2. In Apple Developer Portal, navigate to **Certificates, Identifiers & Profiles**.
3. Under **Identifiers**, select **App IDs**.
4. Click the Add button (**+**) in the upper-right corner.
5. Create a name.
6. Select **Explicit App ID** and enter the App ID provided within the Nintex Mobile Enterprise Portal



App ID Suffix

● **Explicit App ID**

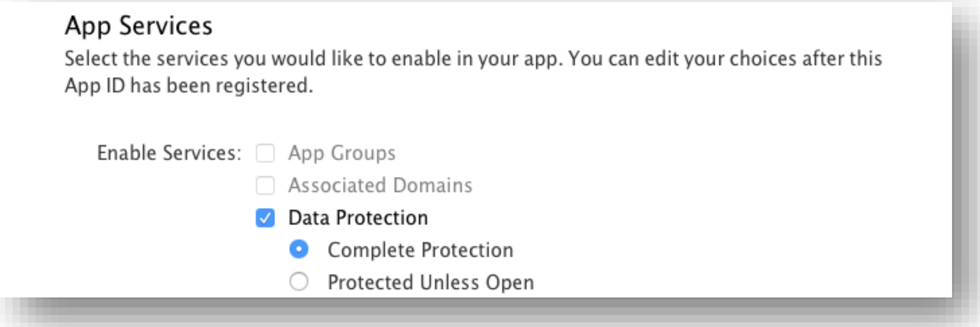
If you plan to incorporate app services such as Game Center, In-App Purchase, Data Protection, and iCloud, or want a provisioning profile unique to a single app, you must register an explicit App ID for your app.

To create an explicit App ID, enter a unique string in the Bundle ID field. This string should match the Bundle ID of your app.

Bundle ID: com.nintex.mobile.blancoff6bf94e03d34166a09ed25150c15470

We recommend using a reverse-domain name style string (i.e., com.domainname.appname). It cannot contain an asterisk (*).

7. Under **App Services**, enable **Data Protection** and the **Complete Protection**.



App Services

Select the services you would like to enable in your app. You can edit your choices after this App ID has been registered.

Enable Services:
- □ App Groups
- □ Associated Domains
- ☑ Data Protection
  - ● Complete Protection
  - ○ Protected Unless Open

8. Click **Continue** and then **Submit** to complete the App ID registration.

Create a Distribution Certificate

1. In Apple Developer Portal, navigate to **Certificates, Identifiers & Profiles**.
2. Click **All** and then the Add button (**+**) in the upper-right corner.

3. Under **Production**, select **In House and Ad Hoc**.

Creating a Certificate Signing Request

To manually generate a Distribution Certificate, a company must create a Certificate Signing Request (CSR) file on a Mac.

To create a Certificate Signing Request

1. In **Applications**, open **Utilities** and then launch **Keychain Access**.
2. In Keychain Access menu bar, select **Keychain Access** > **Certificate Assistant** > **Request a Certificate from a Certificate Authority.**
3. In the Certificate Information window enter the following:

Field	Text
User Email Address	enter email address
Common Name	create a name for a private key
CA Email Address	leave empty
Request is	select **Saved to disk**

4. Click **Continue**.
5. Create a file name and specify where it will be saved.
6. Click **Save**.

To upload a Certificate Signing Request

1. Return to the Apple Developer Portal and click **Continue**.
2. Click **Choose File** and select the file from the previous step.
3. Click **Generate**.

To download a Certificate Signing Request

- Click **Download** to retrieve the Distribution Certificate.

Installing the Certificate

- Click the certificate file to install to the local keychain.

Exporting the Certificate

Some Mobile Device Management (MDM) solutions require the certificate to be exported.

1. In **Applications**, open **Utilities** folder and then launch **Keychain Access**.
2. Under **Keychains**, select **Login**.
3. Under **Category**, select **Keys**.
4. Select the two profiles created previously with the iOS Developer common name. Do not include the public key.
5. Right click and then click **Export 2 items…**.
6. Use the file format **Personal Information Exchange**.
7. Choose a filename and destination and then click **Save**.
8. Set a password and then click **OK**.

Generate a Mobile Provisioning Profile

In order to re-sign an iOS application, a company must create a Mobile Provisioning Profile through the Apple Developer Portal.

1. Navigate to **Certificates, Identifiers & Profiles**.

2. Choose **Identifiers** and then **Provisioning Profiles**.
3. Choose **Distribution** and then Add button (**+**) in the upper-right corner.
4. Under **Distribution**, select **In House**.
5. Select the matching App ID generated by the Nintex Mobile Enterprise Portal and then click **Continue**.
6. Create a name for the Provisioning Profile and then click **Generate**.
7. Click **Download** to use the Provisioning Profile.

Signing an Enterprise App

Before an iOS enterprise app can successfully be distributed, the app must be signed with the generated Mobile Provisioning Profile. There are a number of alternatives available to sign an iOS enterprise app.

Apple Xcode

Apple provides Apple Xcode development tools to create iOS applications. These tools also allow you to sign iOS application package (.ipa) files with a generated Mobile Provisioning Profile.

Note: Apple only enables re-signing on Mac devices.

For more information on Apple Xcode documentation, see Apple Xcode.

iReSign

iReSign is an open source utility that allows companies to easily resign an IOS application package (.ipa) file with an alternate Mobile Provisioning Profile.

Note: Apple only enables re-signing on Mac devices.

For more information on iReSign documentation, see iReSign.

Mobile Device Management Vendors

Some Mobile Device Management (MDM) vendors provide application re-signing and wrapping features, allowing companies to sign iOS Enterprise Apps within the MDM.

Please refer to the relevant MDM documentation for instructions.

Distribute an Enterprise App

Once an iOS enterprise app has been signed with the company's Mobile Provisioning Profile, it is ready for deployment to devices. There are two ways to distribute iOS enterprise apps.

iTunes

iTunes can be used to deploy a signed app to a connected device.

To distribute an app using iTunes

1. Launch iTunes.
2. Click **File** in the menu bar and then select **Add to Library**.
3. Select the (.ipa) file for distribution.
4. Connect the device to the computer.
5. Select the app under the device's **Apps** tab.
6. Click **Apply**.

Mobile Device Management Vendors

Mobile Device Management vendors allow organisations to distribute iOS Enterprise Apps to registered devices.

Please refer to the relevant MDM documentation for instructions.