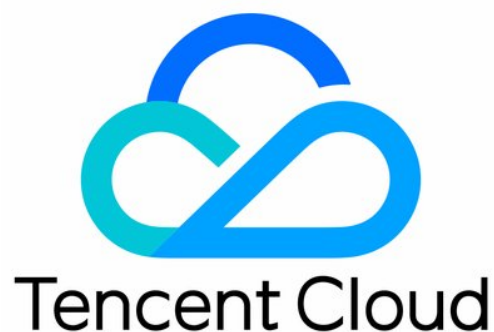


# **Tencent Push Notification Service FAQs Product Documentation**



## Copyright Notice

©2013-2019 Tencent Cloud. All rights reserved.

Copyright in this document is exclusively owned by Tencent Cloud. You must not reproduce, modify, copy or distribute in any way, in whole or in part, the contents of this document without Tencent Cloud's the prior written consent.

## Trademark Notice



All trademarks associated with Tencent Cloud and its services are owned by Tencent Cloud Computing (Beijing) Company Limited and its affiliated companies. Trademarks of third parties referred to in this document are owned by their respective proprietors.

## Service Statement

This document is intended to provide users with general information about Tencent Cloud's products and services only and does not form part of Tencent Cloud's terms and conditions. Tencent Cloud's products or services are subject to change. Specific products and services and the standards applicable to them are exclusively provided for in Tencent Cloud's applicable terms and conditions.

# Contents

## FAQs

iOS

Android

Flutter

Others

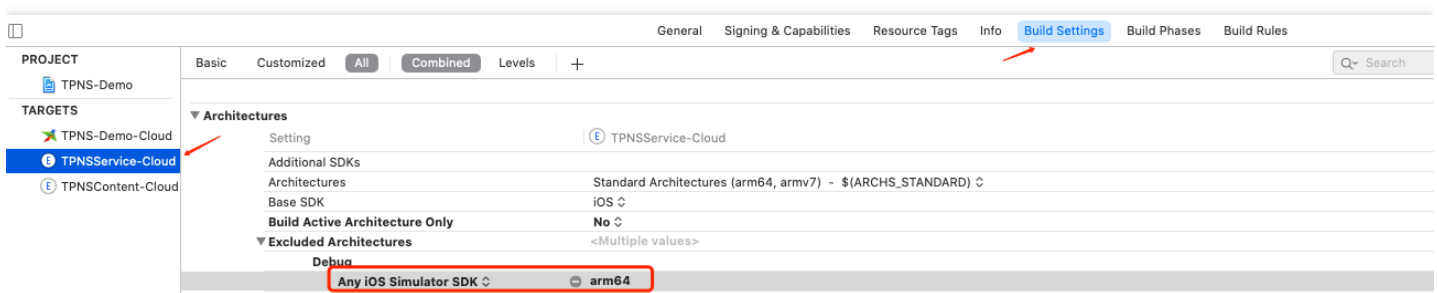
# FAQs

## iOS

Last updated : 2021-03-03 15:29:37

### What should I do if the error `building for iOS Simulator, but linking in object file built for iOS` is reported when the Xcode 12 simulator integrates with the notification extension plugin during the build?

You need to find the extension plugin target, click **Build Settings** > **Excluded Architectures**, and add the arm64 instruction set, as shown in the following figure:



### What should I do if push certificate upload failed in the TPNS console?

Convert the .p12 file of the push certificate into a .pem file and troubleshoot as follows:

1. Open the Terminal and go to the .p12 file directory.
2. Run the following command to generate a certificate ( `apns-dev-cert` is the name of the sample push certificate, which should be replaced with the name of your certificate).

```
openssl pkcs12 -clcerts -nokeys -out apns-dev-cert.pem -in apns-dev-cert.p12
```

3. Enter the password of the .p12 file.
4. Run the following command to convert the .pem certificate into text:

```
openssl x509 -in apns-dev-cert.pem -inform pem -noout -text
```

5. Check whether the certificate environment and corresponding Bundle ID match the application as shown below:

```
Certificate:
Data:
  Version: 3 (0x2)
  Serial Number: 2184588818800376199 (0x1e5137f4cce2f987)
  Signature Algorithm: sha1WithRSAEncryption
  Issuer: C=US, O=Apple Inc., OU=Apple Worldwide Developer Relations, CN=Apple Worldwide Developer Relations Certification Authority
  Validity
    Not Before: Jul 15 11:05:45 2019 GMT
    Not After : Jul 14 11:05:45 2020 GMT
  Subject: UID=com.tencent.teg.duoduo, CN=Apple Development IOS Push Services: com.tencent.teg.duoduo OU=299MFJ63GP, C=CN
  Subject Public Key Info:
    Public Key Algorithm: rsaEncryption
```

## What should I do if an empty notification cannot pop up on devices on iOS 10 or below?

The `content` field cannot be empty if the RESTful API is called for push; otherwise, the notification will not pop up on devices on **iOS 10 or below**.

## Does TPNS support .p8 certificates?

A .p8 certificate has security risks. Although it is valid longer than a .p12 certificate, it has a wider push permission and scope. If leaked, it may cause more severe consequences. TPNS recommends you use .p12 certificates to manage the push services of your applications separately.

## What should I do if the simulator for TPNS SDK v1.2.5.4 or below prompts that the `XGForFreeVersion` symbol cannot be found?

SDK v1.2.5.4 or below only supports debugging on real devices. If you need to use the simulator for debugging, please upgrade to the latest version.

## Why can't push messages be received?

Message push involves various associated modules, and exception in any steps can lead to message delivery failure. Below are the most common issues:

### Client troubleshooting

- Check the notification setting of the device  
Please go to **Notifications > App name** to check whether your app has enabled message push.
- Check the network setting of the device  
If there is a network issue, the client may fail to obtain the message-receiving token when registering for APNs. As a result, TPNS cannot be used to push messages to specified devices.

If the device is not connected to the Internet, it cannot receive the message, even if the client has correctly obtained the token, registered it with the TPNS backend, and the TPNS server has successfully delivered the message. The message might be received if the device reconnects to the Internet within a short time, as APNs will retain the message for some time and deliver it again.

Check the SDK integration. After the SDK is integrated, please make sure that it can get the device token used to receive messages. For more information, please see [iOS SDK Integration Guide](#).

### Server troubleshooting

- APNs server problem

The TPNS server sends a message to an iOS device via APNs. If APNs fails, the TPNS server will fail to request APNs to deliver the message to the device.

- TPNS push server problem

The TPNS server achieves message delivery through the collaboration of multiple feature modules. If an exception occurs in any of these modules, message push will fail.

### Push certificate troubleshooting

When the TPNS server requests APNs to deliver the message, it needs to use two required parameters: the message push certificate and the device token. When pushing the message, please make sure that the message push certificate is valid. For more information on how to configure the message push certificate, please see [Acquisition of Push Certificate](#).

### Why does account/tag binding or unbinding not work?

When SDK APIs are used to bind or unbind accounts/tags, the TPNS server needs about 10 seconds to sync data.

### What do I do if the error `no valid 'aps-environment' entitlement string found for application` is reported in terminal?

Check whether the bundle ID configured in the Xcode project matches the configured provisioning profile, and whether the provisioning profile corresponding to the app has been configured with the message push capability.

### How does the client play the custom push message audio?

First, on the device development side, put the audio file in the `bundle` directory.

- If you use the TPNS console to create a push, enter the audio file name in **Advanced Settings** (the full path of the audio file is not required).
- If you use RESTful APIs, set the `sound` parameter to the name of the audio file (the full path of the audio file is not required).

### Does iOS support offline retention of push messages?

No. When the TPNS server sends a message to APNs, if APNs finds that the device is not online, it will retain the message for a while. However, the duration of the retention is not clear.

## Why is arrival data unavailable for iOS?

- OS versions earlier than iOS 9.x do not provide an API to listen to the arrival of messages at the device. Therefore, the arrival data can be collected.
- OS versions later than iOS 10.0 provide the Service Extension API, which can be called by the client to listen to the arrival of messages.

## How do I create silent push with the TPNS server SDK?

Set `content-available` to `1` and do not use `alert` , `badge` , or `sound` .

## What should I do if `DeviceToken` is not returned occasionally for registration or APNs' request for token fails in the development environment of iOS?

This problem is caused by instability of APNs. You can fix it in the following ways:

1. Insert a SIM card into the phone and use the 4G network.
2. Uninstall and reinstall the application, restart the application, or shut down and restart the phone.
3. Use a package for the production environment.
4. Use another iPhone.

## How do I expand the testing scope on iOS if the number of testing devices is limited?

1. Enterprise signature certificate

Apply for enterprise signature and push certificates and release your application as follows:

Use the enterprise signature certificate to build and release your app. Testers can download and install the application through the dedicated enterprise channel.

2. App Store signature certificate

Use the current push certificate released on App Store as follows:

Release the preview version in TestFlight: upload the IPA package to [App Store Connect](#), use TestFlight to create a beta version, and set the list of testers (Apple IDs) for the specified version in TestFlight. Testers can download and install your application through the official TestFlight application on App Store.

## For iOS, how do I configure to change the badge number only without displaying the message?

When creating a push, you can use the API to specify the notification bar message type, leave the title empty, and only set `badge_type` . For more information, please see [Push API](#).

See below as an example:

```
{
  "platform": "ios",
  "audience_type": "token",
  "environment": "dev",
  "token_list": [
    "05a8ea6924590dd3a94480fa1c9fc8448b4e"],
  "message_type": "notify",
  "message": {
    "ios": {
      "aps": {
        "badge_type": -2
      }
    }
  }
}
```

### What should I do if my application reports the error `Crash: you can't call -sendResponse: twice nor after encoding it ?`

If your application integrates TPNS SDK for iOS (1.2.7.2-1.2.5.4), uses the **Recall** feature of TPNS, and implements the following system callback:

```
- (void)application:(UIApplication *)application didReceiveRemoteNotification:(NSDictionary *)userInfo fetchCompletionHandler:(void (^)(UIBackgroundFetchResult))completionHandler
```

Then you may encounter this error. You can use the **Override** feature to process sent messages.



# Android

Last updated : 2021-01-20 11:09:25

## How do I disable the keep-alive feature of TPNS?

TPNS enables the keep-alive feature by default. Please call the following API in `onCreate` of `Application` or `LauncherActivity` during application initialization and pass in a `false` value:

```
XGPushConfig.enablePullUpOtherApp(Context context, boolean pullUp);
```

If you use the automatic gradle integration method, please configure the following node under the tag of the `AndroidManifest.xml` file of your application, where `xxx` is a custom name. If you use manual integration, please modify the node attributes as follows:

```
<!-- Add the following node to the AndroidManifest.xml file of your application, where xxx is a custom name: -->
<!-- To disable the feature of keep-alive with TPNS application, please configure -->
<provider
    android:name="com.tencent.android.tpush.XGPushProvider"
    tools:replace="android:authorities"
    android:authorities="application package name.xxx.XGVIP_PUSH_AUTH"
    android:exported="false" />
```

If the following log is printed in the console, the session keep-alive feature has been disabled:

```
I/TPush: [ServiceUtil] disable pull up other app .
```

## Why does device registration fail?

- Data sync for newly created applications will take about one minute. During this period, the registration may return the error code 20. You can try again later.
- **Incorrect parameter:** check whether the `Access ID` and `Access Key` are correctly configured. Common errors are that the `Secret key` is misused or the `Access Key` contains spaces.
- **Registration error:** if the console returns an error code such as 10004, 10002, or 20, please see [SDK for Android Error Codes](#).
- **No callback after registration:** check the current network condition. We recommend you use 4G network for testing. Wi-Fi used by many users may have insufficient network bandwidth.
- **Nubia phones:** models released in the second half of 2015 and 2016 cannot be registered, including Nubia Z11 series, Nubia Z11S series, and Nubia Z9S series.

## Why can't pushes be received after registration succeeded?

Please perform automated troubleshooting with the troubleshooting tool as instructed [here](#). General errors include the following:

- Please check whether the current application package name is the same as that entered when TPNS is registered, and if not, we recommend you enable multi-package name push.
- Check whether the network is exceptional on the phone and switch to 4G network for testing.
- TPNS push includes **notification bar message** and **in-app message** (passthrough message). A notification bar message can be displayed on the notification bar, while an in-app message cannot.
- Confirm that the phone is in normal mode. Some phones may have restrictions on network and activity of the backend TPNS process when in Low Power or Do Not Disturb mode.
- Check whether the notification bar permission is granted on the phone. On some OPPO and Vivo phones, the notification bar permission has to be granted manually.

### What is the TPNS channel?

- The TPNS channel is a channel built by TPNS. It can deliver messages only when the TPNS service is online (maintaining a persistent connection with the TPNS backend server). Therefore, the value of the "Sent To" metric of the TPNS channel is generally lower than that of other vendor channels.
- If you need to implement offline push, we recommend you integrate a vendor channel. For more information, please see [Huawei Channel v5 Integration](#).

### Why pushes can't be received after the application is closed?

- At present, third-party push services cannot guarantee that the pushes can be received after the application is closed. This problem is due to the restrictions of the mobile phone's custom ROM on the TPNS service. All pushes through the TPNS channel are on the basis that the TPNS service can maintain a persistent connection with the TPNS backend server. After the service is terminated, whether it can be restarted depends on the system settings, security programs, and user operations.
- After the TPNS service is disconnected from the TPNS server, messages delivered to the device will become offline messages, which can be retained for up to 72 hours. If there are multiple offline messages, only the three latest ones can be retained on each device. If messages are pushed after the application is closed, please check whether the `XGPushManager.unregisterPush¥` (this¥) API has been called if the messages cannot be received when the application is launched again.
- If you have already integrated a vendor channel, but pushes still cannot be received offline, please use the [troubleshooting tool](#) to check whether the token has been successfully registered with the vendor, and if not, please troubleshoot as instructed in [Troubleshooting Vendor Channel Registration Failures](#).

- QQ and WeChat are in the system-level application allowlist, and the relevant service will not be terminated after the application is closed, so the user can still receive messages after closing the application, as the relevant service survives on the backend.

## I installed Huawei Mobile Service on a non-Huawei phone and integrated the TPNS SDK with my application, but this caused Huawei Push and other components to fail. How do I fix it?

Starting from TPNS SDK v1.1.6.3, in order to avoid the situation **where push services of other vendors auto-start and transfer user data on the background on a phone**, push service components of other vendors will be disabled on the phone.

Huawei uses some public components for different features such as account, game, and push. Disabling the push component by TPNS may make other service features unable to start on a non-Huawei phone. If you need to turn off this disablement feature, you can configure the following: Add the following node configuration under the `application` tag of the `AndroidManifest.xml` file, uninstall the application, and reinstall it.

```
<meta-data
  android:name="tpns-disable-component-huawei-v2"
  android:value="false" />
<meta-data
  android:name="tpns-disable-component-huawei-v4"
  android:value="false" />
```

## How do I set the message click event?

TPNS recommends using Intent for redirection (note: the SDK supports click events by default upon message click, and the corresponding homepage will be opened. If the redirection operation is set in `onNotificationClickedResult`, it will conflict with the custom redirection specified in the console/API, resulting in failure of the custom redirection).

### Guide for redirection through Intent:

You need to configure the page to redirect to in the client app's manifest:

- For example, if you want to redirect to the page specified by `AboutActivity`, use the following sample code:

```
<activity
  android:name="com.qq.xg>AboutActivity"
  android:theme="@android:style/Theme.NoTitleBar.Fullscreen" >
  <intent-filter >
    <action android:name="android.intent.action.VIEW" />
    <category android:name="android.intent.category.DEFAULT"/>
```

```

<!-- Specify your complete scheme by customizing the content of the data block. According to your
configuration, a URL identifier in the format of "semantic name://host name/path name" will be fo
rmed -->
<!-- To avoid conflicts with the redirect destination pages of other applications, you can use fi
elds that can uniquely identify the application for configuration, such as those with the applica
tion name or application package name -->
<data
android:scheme="Semantic name"
android:host="Host name"
android:path="/Path name" />
</intent-filter>
</activity>

```

- If you use the TPNS Console to set the intent for redirect, enter in the following way:

- If you use a server SDK, you can set the intent for redirect as follows (with the SDK for Java as an example):

```
action.setIntent("xgscheme://com.tpens.push/notify_detail");
```

- If you want to bring parameters such as `param1` and `param2`, you can set as follows:

```
action.setIntent("xgscheme://com.tpens.push/notify_detail?param1=aa&param2=bb");
```

**Get the parameters on the device:**

1. In the `onCreate` method of the page you specify for redirect to, add the following code:

```
Uri uri = getIntent().getData();
if (uri != null) {
String url = uri.toString();
String p1= uri.getQueryParameter("param1");
String p2= uri.getQueryParameter("param2");
}
```

2. If the parameters passed in contain special characters such as # and &, they can be parsed by using the following method:

```
Uri uri = getIntent().getData();
if (uri != null) {
String url = uri.toString();
UrlQuerySanitizer sanitizer = new UrlQuerySanitizer();
sanitizer.setUnregisteredParameterValueSanitizer(UrlQuerySanitizer.getAllButNullLegal());
sanitizer.parseUrl(url);
String value1 = sanitizer.getValue("key1");
String value2 = sanitizer.getValue("key2");
Log.i("TPNS" , "value1 = " + value1 + " value2 = " + value2);
}
```

**What notification event callbacks do vendor channels support?**

Callback	Arrival Callback	Click Callback
Mi	Supported	Supported
Meizu	Supported	Supported
FCM	Supported	Supported
Huawei	Not supported	Supported
OPPO	Not supported	Supported
Vivo	Not supported	Supported

**⚠ Note :**

Click callback of vendor channels is supported by SDK v1.2.0.1 and above. Legacy versions only supports this feature for Huawei, Mi, Meizu, and Vivo.

## How do I fix the problem of null other push Token that occurs during debugging after my application is integrated with a vendor channel?

A log similar to the following one is found in the application operation logs:

```
[OtherPushClient] handleUpdateToken other push token is : other push type: huawei
```

Then it means that your application failed to register with the vendor channel. You can locate and troubleshoot the problem by getting the return code for vendor channel registration failure. For more information, please see [Troubleshooting Vendor Channel Registration Failures](#).

## Both IM and TPNS are integrated, but there are a lot of vendor conflicts. How do I fix this problem?

Currently, IM uses the vendor jar packages provided by TPNS. Please replace the relevant dependency packages according to the following table to fix the problem.

Push Channel	System Requirements	Condition Description
Mi Push	MIUI	To use Mi Push, add the following dependency: <code>implementation 'com.tencent.tpns:xiaomi:1.2.1.3-release'</code>
Huawei Push	EMUI	To use Huawei Push, add the following dependencies: <ul style="list-style-type: none"> <li><code>implementation 'com.tencent.tpns:huawei:1.2.1.3-release'</code></li> <li><code>implementation 'com.huawei.hms:push:5.0.2.300'</code></li> </ul>
Google FCM Push	Android 4.1 and above	The mobile device needs to have Google Play Services installed and use it outside the Chinese Mainland. Add the following dependency: <code>implementation 'com.google.firebase:firebase-messaging:20.2.3'</code>
Meizu Push	Flyme	To use Meizu Push, add the following dependency: <code>implementation 'com.tencent.tpns:meizu:1.2.1.3-release'</code>
OPPO PUSH	ColorOS	Not all OPPO models and versions support OPPO PUSH. To use OPPO PUSH, add the following dependency: <code>implementation 'com.tencent.tpns:oppo:1.2.1.3-release'</code>
Vivo Push	FuntouchOS	Not all Vivo models and versions support Vivo Push. To use Vivo Push, add the following dependency: <code>implementation 'com.tencent.tpns:vivo:1.2.1.3-release'</code>

## Why can't messages be displayed on the notification bar after arriving at phones on Meizu Flyme 6.0 or below?

1. Manual integration is used on Meizu phones on Flyme 6.0 and below.
2. Automatic integration is used on Meizu phones on Flyme 6.0 and below, and the version of the used TPNS SDK for Android is below 1.1.4.0.

In the above two cases, you need to place an image exactly named `stat_sys_third_app_notify` in the drawable folders with different resolutions. For more information, please see the `flyme-notification-res` folder in the [TPNS SDK for Android](#).

## How do I fix the exception that occurs when I use quick integration in the console?

1. If an exception occurs during integration, set the `"debug"` field in the `tpns-configs.json` file to `true` and run the following command:

```
./gradlew --rerun-tasks :app:processReleaseManifest
```

Then, use the `"TpnsPlugin"` keyword for analysis.

2. Click the **sync projects** icon.



3. Check whether there are relevant dependencies in the external libraries of the project.

```
▶ Gradle: androidx.test:monitor:1.1.1@aar
▶ Gradle: androidx.test:runner:1.1.1@aar
▶ Gradle: androidx.vectordrawable:vectordrawable:1.0.1@aar
▶ Gradle: androidx.vectordrawable:vectordrawable-animated:1.0.0@aar
▶ Gradle: androidx.versionedparcelable:versionedparcelable:1.0.0@aar
▶ Gradle: androidx.viewpager:viewpager:1.0.0@aar
▶ Gradle: com.google.code.findbugs:jsr305:2.0.1@jar
▶ Gradle: com.squareup:javawriter:2.1.1@jar
▶ Gradle: com.tencent.jg:jg:1.1@aar
▶ Gradle: com.tencent.tpns:huawei:1.1.5.2-release@aar
▶ Gradle: com.tencent.tpns:meizu:1.1.5.2-release@aar
▶ Gradle: com.tencent.tpns:oppo:1.1.5.2-release@aar
▶ Gradle: com.tencent.tpns:tpns:1.1.5.2-release@aar
▶ Gradle: com.tencent.tpns:vivo:1.1.5.2-release@aar
▶ Gradle: com.tencent.tpns:xiaomi:1.1.5.2-release@aar
▶ Gradle: javax.inject:javax.inject:1@jar
▶ Gradle: junit:junit:4.12@jar
```

## How do I convert Android extension library v4 to AndroidX?

Add the following property to the `gradle.properties` file of the AndroidX project:

```
android.useAndroidX=true
android.enableJetifier=true
```

### Note :

- `android.useAndroidX=true` indicates to enable `AndroidX` for the current project.
- `android.enableJetifier=true` indicates to migrate the dependency package to `AndroidX`.

## What should I do if a vendor channel's push SDK "transfers information over HTTP in plaintext"?

After you integrate the push services of various vendor channels, some security detection tools may prompt that "the application transfers information over HTTP in plaintext". The specific HTTP addresses include:



1. Mi Push SDK: <http://new.api.ad.xiaomi.com/logNotificationAdActions>,  
<http://resolver.msg.xiaomi.net/psc/?t=a>
2. Meizu Push SDK: <http://norma-external-collect.meizu.com/android/exchange/getpublickey.do>,  
<http://norma-external-collect.meizu.com/push/android/external/add.do>

All the above HTTP URLs are from the push SDKs of relevant vendors. The TPNS team is unable to clarify their purposes or control their behaviors but is actively contacting them and promoting the adoption of transfer over HTTPS. Currently, you should evaluate and choose whether to continue to use the above vendors' push services.

# Flutter

Last updated : 2020-12-14 11:39:04

## If Flutter is integrated, how can I get custom parameters upon the cold start of an iOS device?

To get custom parameters, use the `runner->AppDelegate->didFinishLaunchingWithOptions` method of the following APIs:

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Get the message content
    NSDictionary *remoteNotification = [launchOptions objectForKey:UIApplicationLaunchOptionsRemoteNotificationKey];
    // Then process logically based on the message content
}
```

## Why can't the notification click callback be triggered on Android sometimes after Flutter is integrated?

We recommend you call the `XgFlutterPlugin().addEventHandler()` API to set notification callback events as soon as Flutter is initialized in order to ensure the timeliness of callback API settings upon cold start of the application. In addition, please add all the required callbacks in one call of the `XgFlutterPlugin().addEventHandler()` API, as multiple calls will overwrite them in turn and invalidate callbacks added previously. For more information, please refer to the API call method in the `example/lib/main.dart` file in the project directory.

# Others

Last updated : 2021-02-07 11:40:39

## How do I use the `AccessID` , `AccessKey` , and `SecretKey` of a TPNS application?

- `AccessID`: unique identifier of the TPNS application. Use cases: 1. SDK integration; 2. Authentication signature generation during RESTful API call.
- `AccessKey`: client authentication key of the TPNS application. Use cases: SDK integration.
- `SecretKey`: server authentication key of the TPNS application. Use cases: authentication signature generation during RESTful API call.

## Why is my application rejected after the review when I publish it on Huawei AppGallery?

Please download the official Huawei HMS SDK and copy all files and subdirectories in the `assets` directory to that in your application project. If the `assets` directory does not exist in the application project, create one.

## Why can't pushes be received on a Nubia phone?

TPNS does not support Nubia phones released after 2015, as the new version of Nubia operating system has a super power-saving feature, which will kill background processes very quickly and prevent TPNS Service from being started, resulting in failures of registration with Nubia phones.

## Why can't pushes be received in an iOS build production environment?

1. Check whether the production environment meets the following testing conditions: the application is the ad-hoc/App Store build (with the release certificate "Production"), and the release certificate is uploaded and successfully verified.
2. Check whether the `bundle id` configured in the Xcode project matches the configured `Provision Profile` file and whether the `Provision Profile` file corresponding to the application has been configured with the message push capability.
3. Check whether the environment specified by the `aps-environment` field in the `embedded.mobi leprovision` file is correct.

## Why can't pushes be received in the production environment?

Check whether the production environment meets the following testing conditions: the application is a build packaged with the ad-hoc certificate or the release certificate ("Production") and uploaded to App Store.

## Why is the iOS token invalid?

- The system was logged out or the application was uninstalled.
- The user installed the application on a new device.
- The user restored the device from a backup.
- The user reinstalled the operating system.
- Other system-defined events. (After calling the `unregisterNotification` API, register the notification again and clear device data and settings.)

## Why can only one push message be displayed on a device that has integrated the Mi channel?

According to the documentation at Xiaomi official website, the notification bar only displays one push message by default. If you want multiple messages to be displayed in the notification bar, you need to set a unique `notify_id` for different messages (a new notification bar message with the same `notify_id` will override the previous one). The parameter at TPNS official website is `n_id`.

## Is there a limit on the number of messages displayed in the notification bar?

There is no limit on the number of notification bar messages that a phone can receive and display. The reasons for not displaying may include:

- The notification bar on a Mi phone displays the latest message by default. If you want multiple messages to be displayed, you need to set a unique `n_id` for different messages.
- The message broadcast is blocked by a phone manager application.
- A Meizu phone has a message box, and uncommon messages will go directly to it. Please view the messages there.

## How do I set a custom ringtone?

1. In the console: select **Push Notification > Advanced Settings > Reminder Mode > Sound > Customize** (for Android, select the ringtone file in the `raw` directory, which does not need the extension, such as `xg_ring`. For iOS, select the ringtone file in the `bundle` directory, which requires the extension, such as `xg_ring.wav`).
2. In RESTful API v3: for Android, set `ring=1` in the push message body, and set `ring_raw` to the name of the specified ringtone file without the extension in the `raw` directory of the Android project. For iOS, set the sound in the push message body to the name of the specified ringtone file with the extension in the `bundle` directory of the project.

### Note :

If the client has integrated with a vendor channel, due to the limitations of Huawei and Meizu, a custom ringtone file is not allowed on their phones, and the system sound will be used by default. Currently, a custom ringtone can be used on Mi phones.

## How do I adapt to small icons?

- ROMs running native Android 5.0 or above will process the small icon of an application and add a layer of color if `target_sdk` is greater than or equal to 21, causing the icon to be gray.
- If you want to display it as colored, you can set `target_sdk` to below 21. If you don't want `target_sdk` to be below 21, you can rename a small transparent background .png image to `notification_icon.png` (the filename must be unique) and place it in the `drawable` directory; in this way, the small icon will be displayed as gray (but shaped).
- Starting from TPNS SDK for Android v1.2.2.0, the `notification_icon.png` small icon resource will only take effect directly on Google Pixel phones by default. To achieve such small icon effect for custom notifications on other phones, you need to specify the resource filename (without the extension) as the `message.android.small_icon` field in the push API. In addition, the custom notification small icon supports solid colors by specifying a decimal value of an RGB color as the `message.android.icon_color` field in the push API.

Below is an example of push API fields, where `icon_color: 123456` indicates the RGB color 01e240:

```
{
  "message": {
    "android": {
      "small_icon": "notification_icon",
      "icon_color": 123456
    }
  }
}
```

### **Note :**

- The small icon must be a PNG image with an alpha channel.
- The background must be transparent.
- Do not leave too much padding around the icon.
- We recommend you use an image with dimensions of 46x46, as smaller images will be blurry, while larger images will be automatically scaled down.

## Can an application still receive push messages after it is closed or its process is ended?

- The TPNS channel mainly uses TPNS Service to push and receive messages. When the process is ended, TPNS Service will also be ended. Messages be received and pushed only after TPNS Service is restarted. If another application connected to TPNS is opened on the phone, messages can be

received and pushed by using TPNS Service of that application. However, TPNS Service channel sharing is also subject to the phone's ROM, and 100% success rate cannot be guaranteed.

- Vendor channels can receive push messages even after the application process is ended.

### What should I do if Android v4.4.4 reports a compiling error?

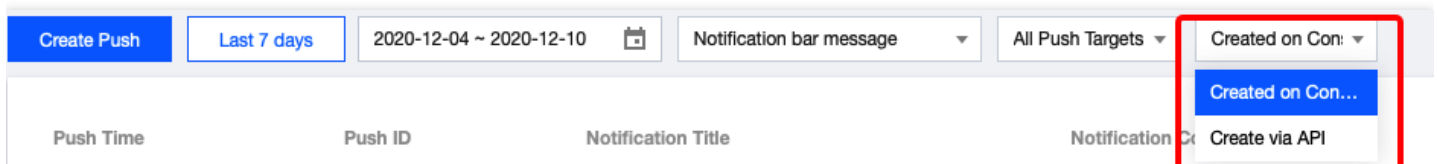
If the number of loaded methods in the project exceeds 65,000, please create subpackages for the project.

### Why can't callback information be received for device registration?

- A vendor channel's callbacks are returned by the vendor server.
- You can check whether broadcast is blocked by any security application.

### Why can't I find the records of pushes created through API?

Log in to the [TPNS console](#) and select **Created via API** on the **Push Management > Task List** page to view the records of pushes created through API.



### If an account changes its bound device, what will happen when a message is sent to this account?

The currently bound device (B) can receive the push, but the original bound device (A) cannot, as only the last device bound to the account can receive pushes.

### What should I do if I specified to open an Activity page but it often cannot be redirected to normally?

On some phones, redirection upon click on the notification bar may experience permission issues. Solution: in `androidManifest.xml`, add `android:exported="true"` to the `Activity` to be opened.

### What is the order in which multiple push messages are received after a user goes back online?

The messages are in ascending order by message ID. The client will also receive messages in this rule; therefore, the order in which the messages are received is the same as that in which they are sent.

### If a past time is selected for a scheduled push, will the push be sent?

Yes. If a past time is selected, the system will send the push immediately.

### **Can the registration method be created in a thread?**

The registration method can be called anywhere, but `ApplicationContext` needs to be passed.