

Ant Technology

Device ID
User Guide

Document Version: 20250303



Legal disclaimer

Ant Group all rights reserved ©2022.

No part of this document shall be excerpted, translated, reproduced, transmitted, or disseminated by any organization, company, or individual in any form or by any means without the prior written consent of Ant Group.

Trademark statement



蚂蚁集团 and other trademarks related to Ant Group are owned by Ant Group. The third-party registered trademarks involved in this document are owned by the right holder according to law.

Disclaimer

The content of this document may be changed due to product version upgrades, adjustments, or other reasons. Ant Group reserves the right to modify the content of this document without notice and the updated versions of this document will be occasionally released through channels authorized by Ant Group. You must pay attention to the version changes of this document as they occur and download and obtain the latest version of this document from Ant Group's authorized channels. Ant Group does not assume any responsibility for direct or indirect losses caused by improper use of documents.

Document conventions

Style	Description	Example
 Danger	A danger notice indicates a situation that will cause major system changes, faults, physical injuries, and other adverse results.	 Danger: Resetting will result in the loss of user configuration data.
 Warning	A warning notice indicates a situation that may cause major system changes, faults, physical injuries, and other adverse results.	 Warning: Restarting will cause business interruption. About 10 minutes are required to restart an instance.
 Notice	A caution notice indicates warning information, supplementary instructions, and other content that the user must understand.	 Notice: If the weight is set to 0, the server no longer receives new requests.
 Note	A note indicates supplemental instructions, best practices, tips, and other content.	 Note: You can use Ctrl + A to select all files.
>	Closing angle brackets are used to indicate a multi-level menu cascade.	Click Settings> Network> Set network type .
Bold	Bold formatting is used for buttons, menus, page names, and other UI elements.	Click OK .
Courier font	Courier font is used for commands	Run the <code>cd /d C:/window</code> command to enter the Windows system folder.
<i>Italic</i>	Italic formatting is used for parameters and variables.	<code>bae log list --instanceid</code> <i>Instance_ID</i>
[] or [a b]	This format is used for an optional value, where only one item can be selected.	<code>ipconfig [-all -t]</code>
{ } or {a b}	This format is used for a required value, where only one item can be selected.	<code>switch {active stand}</code>

Table of Contents

1. Device ID	05
1.1. Overview	05
1.2. Integrate Device ID into Android	05
1.3. Integrate Device ID into iOS	05
1.4. FAQ	06

1. Device ID

1.1. Overview

By using the Device ID component, you can easily and conveniently obtain the device IDs, thus ensuring that the applications can safely and efficiently find specific devices.

1.2. Integrate Device ID into Android

The Device ID component can be integrated to the mPaaS in the native AAR mode and the component-based mode.

Prerequisites

- Before you connect the component to the mPaaS based on native AAR mode, [Add mPaaS to your project](#) first.
- If you want to connect the component to the mPaaS based on components, you need to first complete the [Component-based integration procedure](#).

Add the SDK

Native AAR mode

In your project, install the **Device ID** component on the **Component Management (AAR)** page. For more information, see [AAR component management](#).

Component-based mode

In your Portal and Bundle projects, install the **Device ID** component on the **Component Management** page.

For more information, see [Manage component dependencies](#).

Obtain device ID

```
UTDevice
/**
 * Obtain the unique device ID.
 *
 *
 * @return Unique device ID
 */
public static String getUtdid(Context context);
```

1.3. Integrate Device ID into iOS

This guide introduces how to integrate Device ID to iOS client. You can integrate Device ID to iOS client based on native project with CocoaPods.

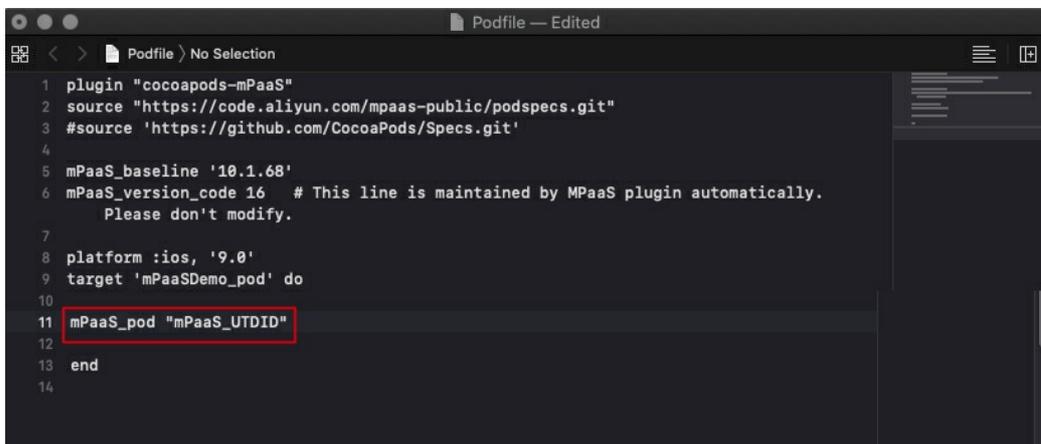
Prerequisites

You have connected your project to mPaaS. For more information, refer to: [Integrate mPaaS based on an existing project and CocoaPods](#).

Add the SDK

Use the cocoapods-mPaaS plug-in to add the SDK.

1. In the Podfile file, add `mPaaS_pod "mPaaS_UTDID"` to add the dependencies of the Device ID component.



2. Run `pod install` to connect the component to the mPaaS.

Use the SDK

After you add the SDK, you can call the following method to obtain the unique device ID:

```
/**
 * Obtain the unique device ID generated by the SDK.
 */
+ (NSString *)deviceId;
```

For more information, see the `MPUtudidInterface.h` file in `MPUTDIDAdapter.framework`.

1.4. FAQ

This topic describes FAQs about UTDID on Android.

Under what circumstances does the value of UTDID remain unchanged and under what circumstances does it not?

Answer:

- After the mobile phone or application is restarted

	Without permissions on the SD card	With permissions on the SD card
The value of UTDID	Unchanged	Unchanged

- After the application is deleted and reinstalled

	Without permissions on the SD card	With permissions on the SD card
--	------------------------------------	---------------------------------

The value of UTDID	Changed	Unchanged
---------------------------	---------	-----------

- After you wipe the data and cache on your device, clear the data in the SD card, and then reinstall the application

	Without permissions on the SD card	With permissions on the SD card
The value of UTDID	Changed	Changed

Why is UTDID changed after I terminate a process?

Cause analysis:

1. UTDID is the ID of an application on a device. Every time when the application is started, the UTDID outside the application is compared against the UTDID inside the application. The UTDID that was generated at an earlier time is used.
2. When the application is installed for the first time, the application cannot obtain the outside UTDID because the application is not authorized to access external storage. In addition, the application does not have an inside UTDID. Therefore, the application generates a UTDID.
3. After you authorize the application to access external storage and restart the application, the UTDID outside the application overwrites the UTDID inside the application. Therefore, the UTDID is changed.

Solution:

1. Make sure that the version of utdid4all is not earlier than V1.1.5.3. A version earlier than V1.1.5.3 is incompatible with advanced versions of Android.
2. This problem occurs only when you install the application for the first time. After you authorize the application to access external storage, the UTDID is not changed when you restart the application.

Note

For Android 6.0 and later, you cannot authorize an application to access external storage only by declaration. When you authorize the application on a specific device model, the authorization page may appear and you must confirm the authorization.

How do I ensure that the ID of a device remains unchanged in Android 10 and later?

As device authorization policies are more and more strict, the UTDID is difficult to remain unchanged on a device. If you need a fixed device ID in specific scenarios, use another way to specify the device ID.