Tencent Cloud Observability Platform

Mobile App Performance Monitoring

Product Documentation

Contents

- Mobile App Performance Monitoring
  - Overview
  - Access Guide
    - Android Use Cases
      - Integration and Initialization
      - Feature Configuration
        - Network Monitoring
        - WebView, JsError, and Web Network Monitoring
        - Crash and ANR Monitoring
        - Lag and Frame Rate Monitoring
        - Startup Monitoring
    - iOS Use Cases
      - Integration and Initialization
  - Operation Guide
    - Crash
    - ANR
    - Latency
    - Startup
    - Network
    - Webview
    - Application Management

Mobile App Performance Monitoring Overview

Last updated：2024-05-14 12:36:06

Overview

Mobile App Performance Monitoring is a comprehensive tool for positioning, detecting app performance, and user experience, automatically analyzing suspicious performance defects in multiple dimensions. It helps you accurately measure the app performance and discover various issues with low cost and high efficiency.

App Monitoring Feature Description

Feature Name	Description
Crash	By aggregating key characteristics of individual crash cases, it facilitates the locating and analysis of the root cause.
Startup	Supports startup metric analysis such as startup duration and slow startup percentage, allowing you to locate and analyze the root cause of slow startups through the slow startup issue list.
Latency	Metrics like smoothness help you analyze the performance of app pages.
ANR	Multi-dimensional restoration of the real experience of online users. It collects ANR issues encountered during the real use of the app and the thread stack information upon the occurrence of the issue to extract key features for clustering.
Network	Supports network problem analysis based on request duration, slow request proportion, network error rate, and other metrics.
WebView	Supports WebView metric analysis based on page load time, slow loading proportion, and JS error rate, and dives into WebView and JSError issues through the issue list.

Data Storage Description

Individual case data (slow startup and requests): stored for 60 days

15-minute metric data (metric analysis interface): stored for 30 days

1-hour metric data (metric analysis interface): stored for 30 days

Access Guide

Android Use Cases

Integration and Initialization

Last updated：2024-05-24 11:54:32

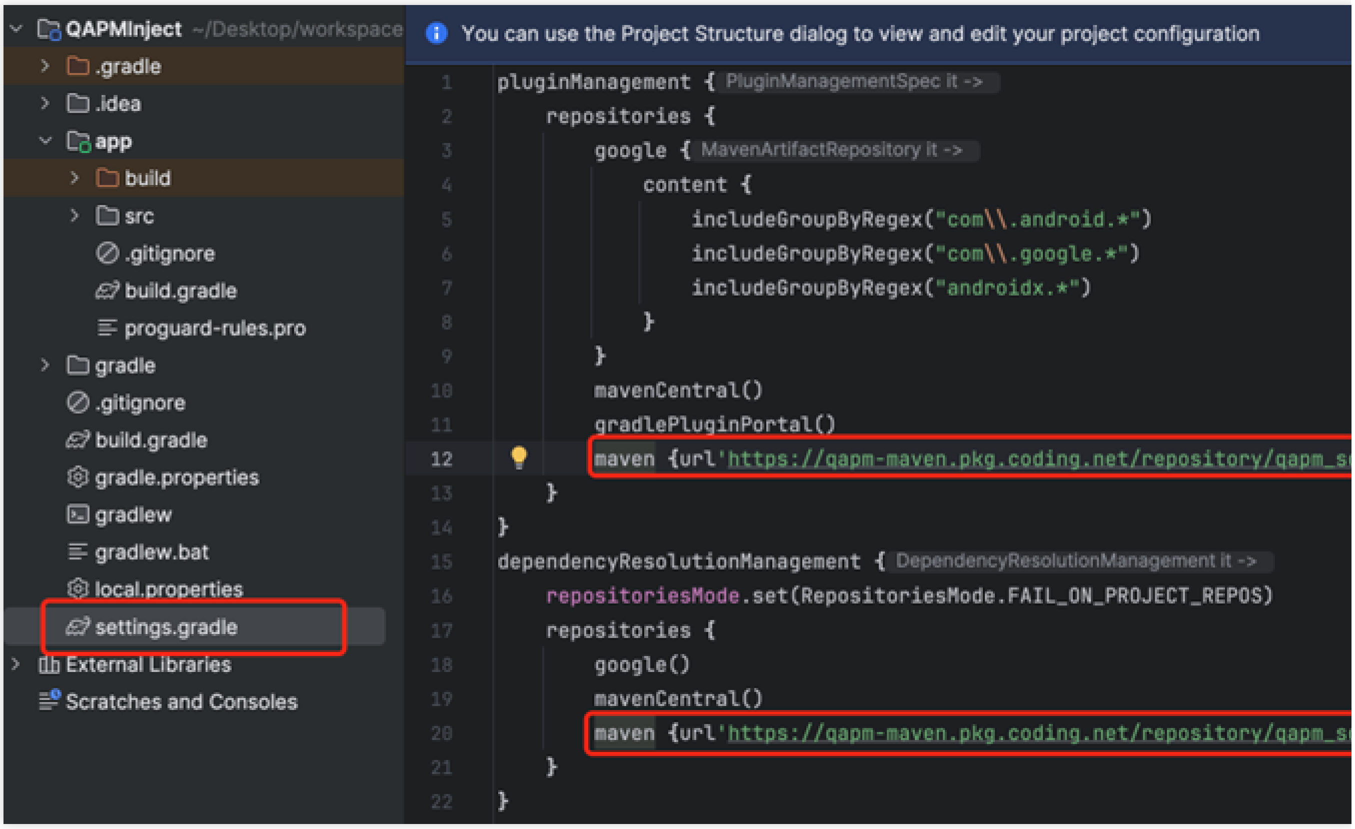
Overview

This document guides you through integrating and initializing with the Android SDK.

Directions

Step 1: Configure Gradle integration.

1. Add the Maven repository source in **settings.gradle**.



You can use the Project Structure dialog to view and edit your project configuration

```
pluginManagement { PluginManagementSpec it ->
    repositories {
        google { MavenArtifactRepository it ->
            content {
                includeGroupByRegex("com\\.android.*")
                includeGroupByRegex("com\\.google.*")
                includeGroupByRegex("androidx.*")
            }
        }
        mavenCentral()
        gradlePluginPortal()
        maven {url'https://qapm-maven.pkg.coding.net/repository/qapm_s
    }
}
dependencyResolutionManagement { DependencyResolutionManagement it ->
    repositoriesMode.set(RepositoriesMode.FAIL_ON_PROJECT_REPOS)
    repositories {
        google()
        mavenCentral()
        maven {url'https://qapm-maven.pkg.coding.net/repository/qapm_s
    }
}
```

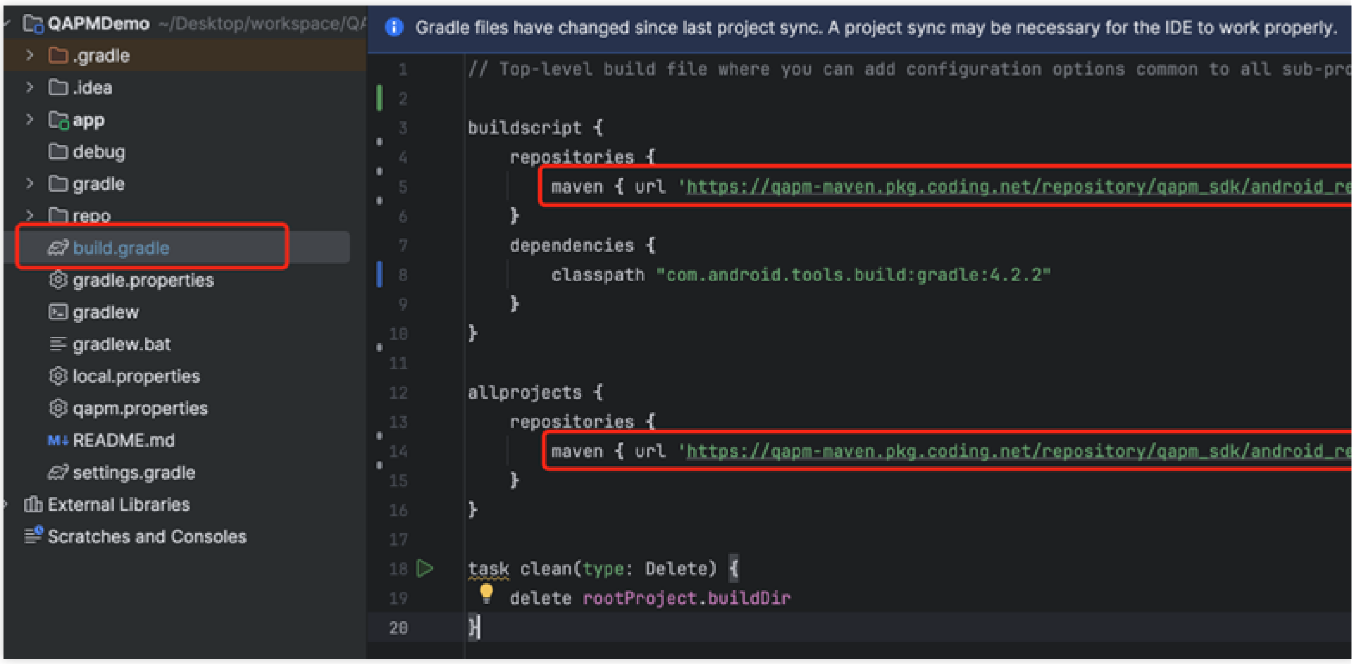
Reference code:

```
pluginManagement {
    ...
    repositories {
      ...
      // Add the following content.
      maven { url 'https://qapm-maven.pkg.coding.net/repository/qapm_sdk/android_
    }
}

dependencyResolutionManagement {
     ...
```

```
        repositories {
          ...
          // Add the following content.
          maven { url 'https://qapm-maven.pkg.coding.net/repository/qapm_sdk/android
        }
    }
```

Note:

If your **Gradle version is below 7.0**, please add the Maven repository source in **the project's build.gradle** as shown below:

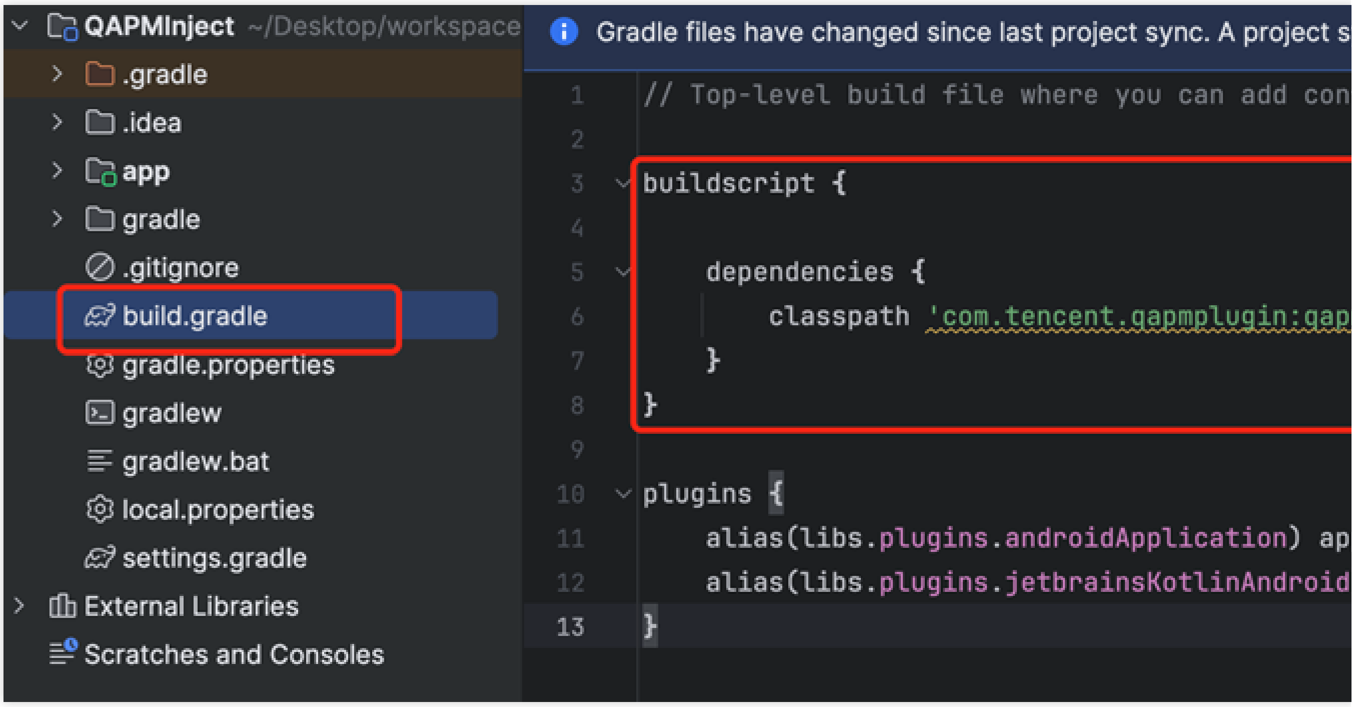


QAPMDemo ~/Desktop/workspace/Q
.gradle
.idea
app
debug
gradle
repo
build.gradle
gradle.properties
gradlew
gradlew.bat
local.properties
qapm.properties
README.md
settings.gradle
External Libraries
Scratches and Consoles

Gradle files have changed since last project sync. A project sync may be necessary for the IDE to work properly.

```
// Top-level build file where you can add configuration options common to all sub-pro

buildscript {
    repositories {
        maven { url 'https://qapm-maven.pkg.coding.net/repository/qapm_sdk/android_re
    }
    dependencies {
        classpath "com.android.tools.build:gradle:4.2.2"
    }
}

allprojects {
    repositories {
        maven { url 'https://qapm-maven.pkg.coding.net/repository/qapm_sdk/android_re
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

2. Add dependencies for the plugin in the **project's build.gradle** file.



Gradle files have changed since last project sync. A project s

```
// Top-level build file where you can add con

buildscript {

    dependencies {
        classpath 'com.tencent.qapmplugin:qap
    }
}

plugins {
    alias(libs.plugins.androidApplication) ap
    alias(libs.plugins.jetbrainsKotlinAndroid
}
```

Reference code:

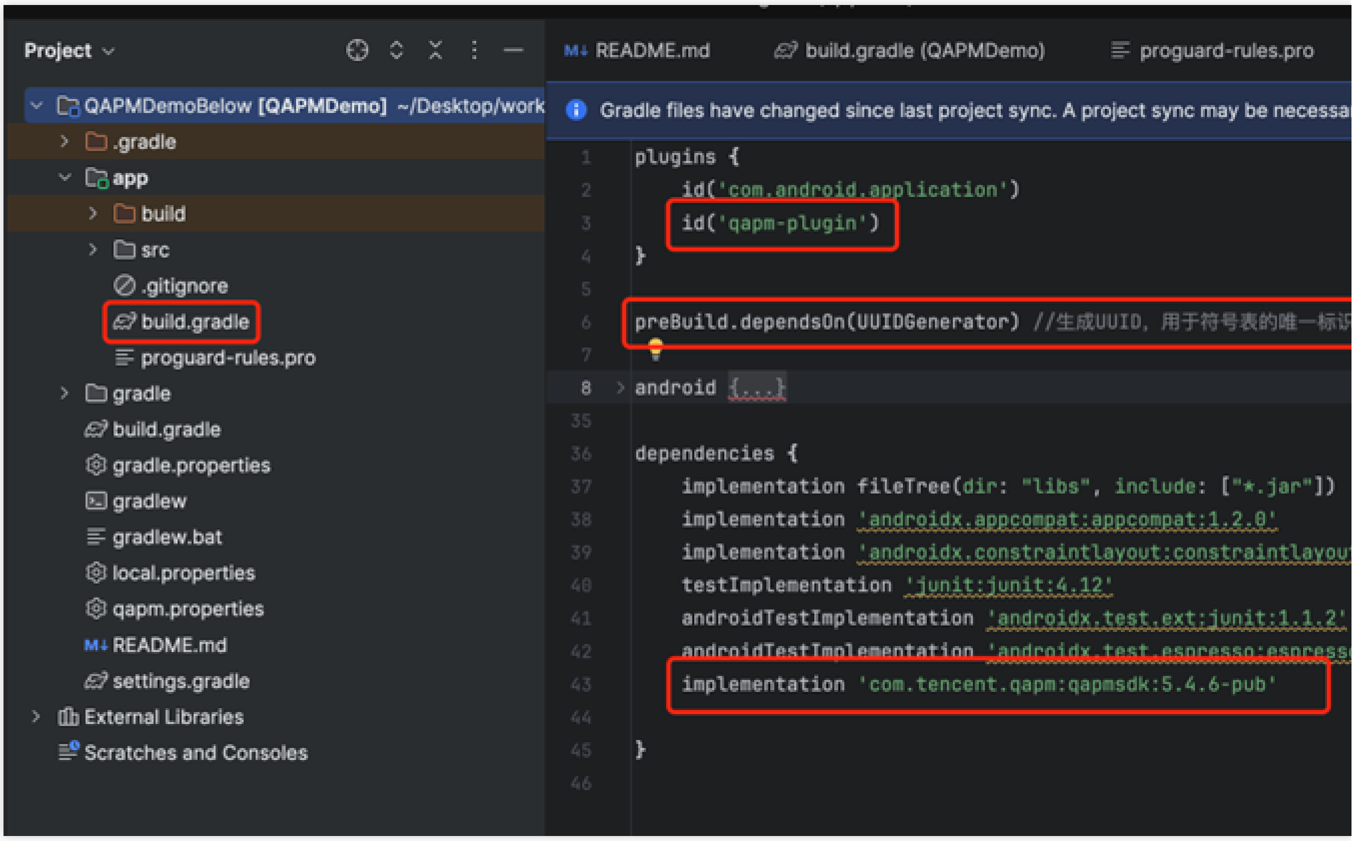
```
}
```

```
buildscript {
       ...
     dependencies {
         ...
         // Add the following content
         classpath 'com.tencent.qapmplugin:qapm-plugin:3.1'
         ...
     }
 }
```

Note:

If your **Gradle version is below 7.4**, change the plugin version to 2.39.

3. Add the following code to the **app's build.gradle** file:



Gradle files have changed since last project sync. A project sync may be necessa

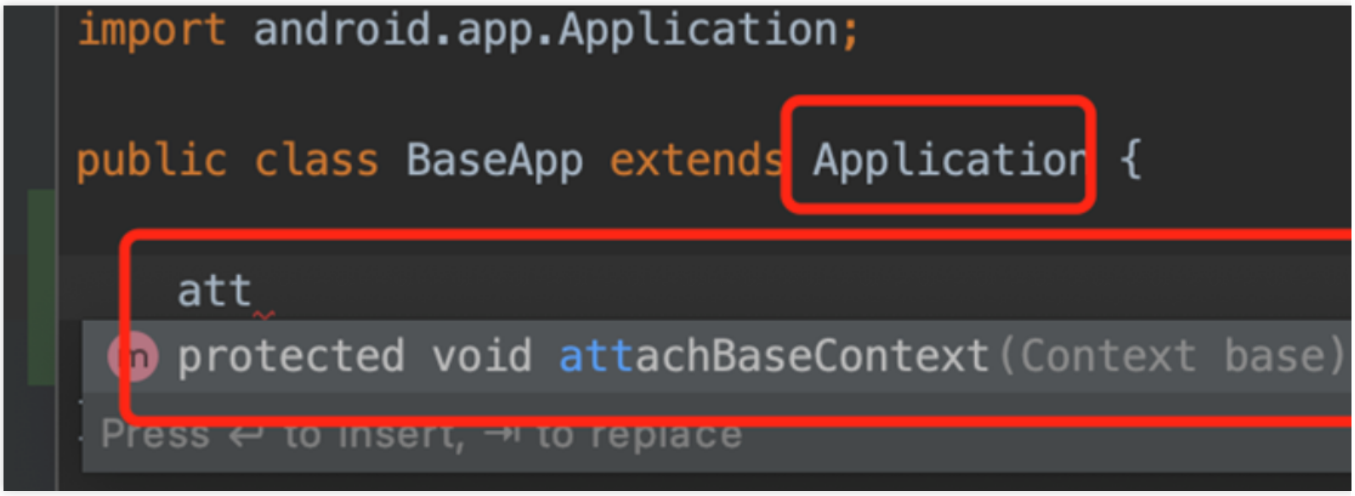
```
plugins {
    id('com.android.application')
    id('qapm-plugin')
}

preBuild.dependsOn(UUIDGenerator) //生成UUID，用于符号表的唯一标识

android {...}

dependencies {
    implementation fileTree(dir: "libs", include: ["*.jar"])
    implementation 'androidx.appcompat:appcompat:1.2.0'
    implementation 'androidx.constraintlayout:constraintlayou
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'androidx.test.ext:junit:1.1.2'
    androidTestImplementation 'androidx.test.espresso:espress
    implementation 'com.tencent.qapm:qapmsdk:5.4.6-pub'

}
```

Reference code:

```
plugins {
    ...
    id('qapm-plugin') // Add the plugin.
    ...
}

...
preBuild.dependsOn(UUIDGenerator) // Generate a unique identifier for subsequent
...

dependencies {
    ...
    // Add qapmsdk dependency.
    implementation 'com.tencent.qapm:qapmsdk:5.4.6-pub'
    ...
}
```

4. Please check if this step needs to be performed by the following content.

In the class where the Application is located, enter attachBaseContext to check if there is an override method. If there is an override method, ignore this step. If not, please proceed to Next.



```
import android.app.Application;

public class BaseApp extends Application {

    att
    protected void attachBaseContext(Context base)
    Press ↩ to insert, ⇥ to replace
```

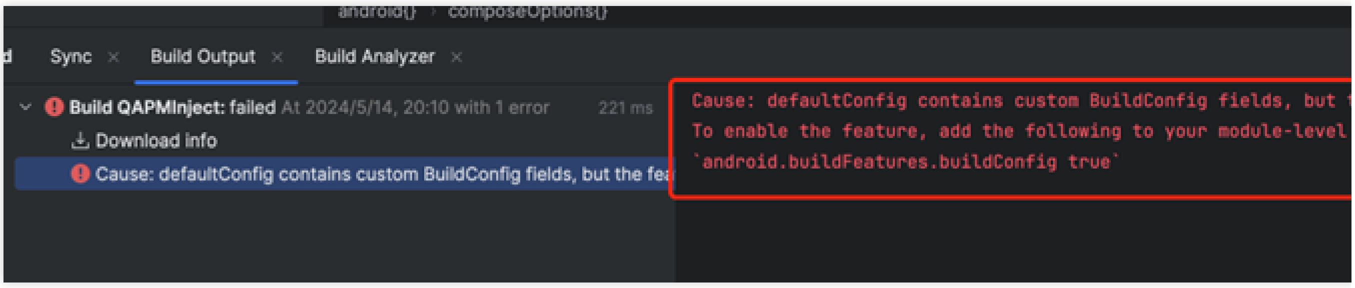
Add the package path of the Application to the following configuration.

Reference code:

```
QAPMPluginConfig {
   // Optional, defaults to empty. In the class where Application is located, enter
   // tinkerApplication = 'com/tencent/qapm/demoApplication'
}
```

5. At this point, you can try to compile. **Compilation-related FAQ** can be referred to for this step.

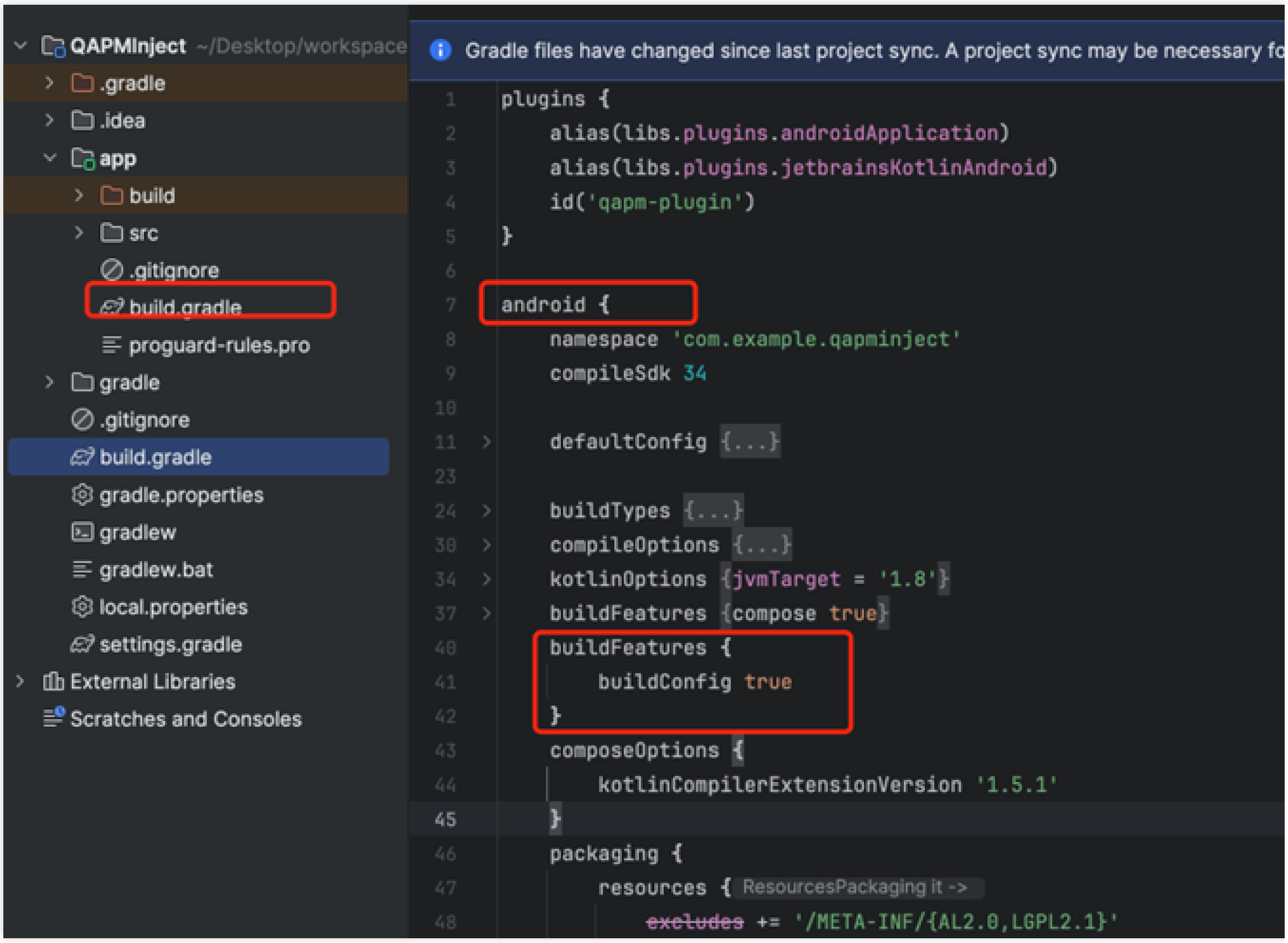
Q1: If an error stating "**feature is disabled**" occurs during compilation, see below:



```
Build QAPMInject: failed At 2024/5/14, 20:10 with 1 error    221 ms
  Download info
  Cause: defaultConfig contains custom BuildConfig fields, but the fea
```

```
Cause: defaultConfig contains custom BuildConfig fields, but t
To enable the feature, add the following to your module-level
`android.buildFeatures.buildConfig true`
```

A1: The plugin needs to dynamically insert properties into BuildConfig. Please add the following code under the **app module's build.gradle** file.



```
QAPMInject ~/Desktop/workspace
  .gradle
  .idea
  app
    build
    src
    .gitignore
    build.gradle
    proguard-rules.pro
  gradle
  .gitignore
  build.gradle
  gradle.properties
  gradlew
  gradlew.bat
  local.properties
  settings.gradle
External Libraries
Scratches and Consoles
```

Gradle files have changed since last project sync. A project sync may be necessary fo

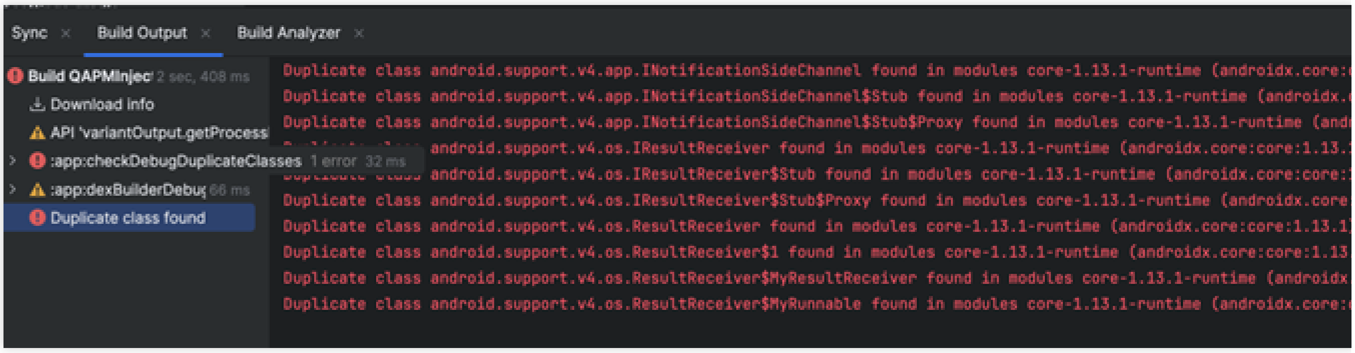
```
plugins {
    alias(libs.plugins.androidApplication)
    alias(libs.plugins.jetbrainsKotlinAndroid)
    id('qapm-plugin')
}

android {
    namespace 'com.example.qapminject'
    compileSdk 34

    defaultConfig {...}

    buildTypes {...}
    compileOptions {...}
    kotlinOptions {jvmTarget = '1.8'}
    buildFeatures {compose true}
    buildFeatures {
        buildConfig true
    }
    composeOptions {
        kotlinCompilerExtensionVersion '1.5.1'
    }
    packaging {
        resources { ResourcesPackaging it ->
            ~~excludes~~ += '/META-INF/{AL2.0,LGPL2.1}'
```

Reference code:

}

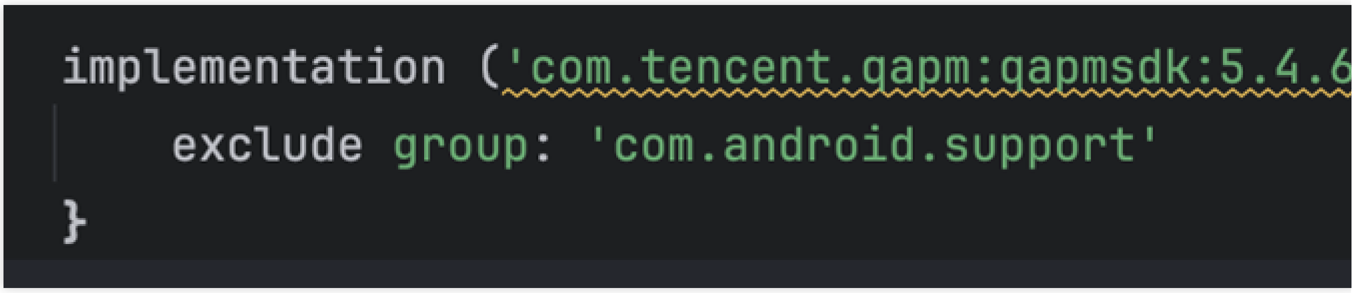
```
android {
    ...
    // Add the following content.
    buildFeatures {
       ...
       buildConfig true
    }
    ...
}
```

Q2: If a "Class Conflict" error occurs during compilation, as follows:


Sync
Build Output
Build Analyzer
Build QAPMInjec 2 sec, 408 ms
Download info
API 'variantOutput.getProcess
:app:checkDebugDuplicateClasses 1 error 32 ms
:app:dexBuilderDebug 66 ms
Duplicate class found

```
Duplicate class android.support.v4.app.INotificationSideChannel found in modules core-1.13.1-runtime (androidx.core:
Duplicate class android.support.v4.app.INotificationSideChannel$Stub found in modules core-1.13.1-runtime (androidx.
Duplicate class android.support.v4.app.INotificationSideChannel$Stub$Proxy found in modules core-1.13.1-runtime (and
Duplicate class android.support.v4.os.IResultReceiver found in modules core-1.13.1-runtime (androidx.core:core:1.13.
Duplicate class android.support.v4.os.IResultReceiver$Stub found in modules core-1.13.1-runtime (androidx.core:core:
Duplicate class android.support.v4.os.IResultReceiver$Stub$Proxy found in modules core-1.13.1-runtime (androidx.core
Duplicate class android.support.v4.os.ResultReceiver found in modules core-1.13.1-runtime (androidx.core:core:1.13.1
Duplicate class android.support.v4.os.ResultReceiver$1 found in modules core-1.13.1-runtime (androidx.core:core:1.13
Duplicate class android.support.v4.os.ResultReceiver$MyResultReceiver found in modules core-1.13.1-runtime (androidx
Duplicate class android.support.v4.os.ResultReceiver$MyRunnable found in modules core-1.13.1-runtime (androidx.core:
```

A2: Qapm and your project use different Android support libraries. Please remove the qapm's android.support library when adding dependencies, as follows:



```
implementation ('com.tencent.qapm:qapmsdk:5.4.6
    exclude group: 'com.android.support'
}
```

Reference code:

```
}
```

```
dependencies {
    ...
  implementation ('com.tencent.qapm:qapmsdk:5.4.6-pub') {
    // Add the following content.
    exclude group: 'com.android.support'
  }
  ...
}
```

Note:

If the qapm-plugin plugin is not used, it will affect the monitoring of the startup and network.

Step 2: Configure parameters.

1. Add the following permissions in AndroidManifest.xml.

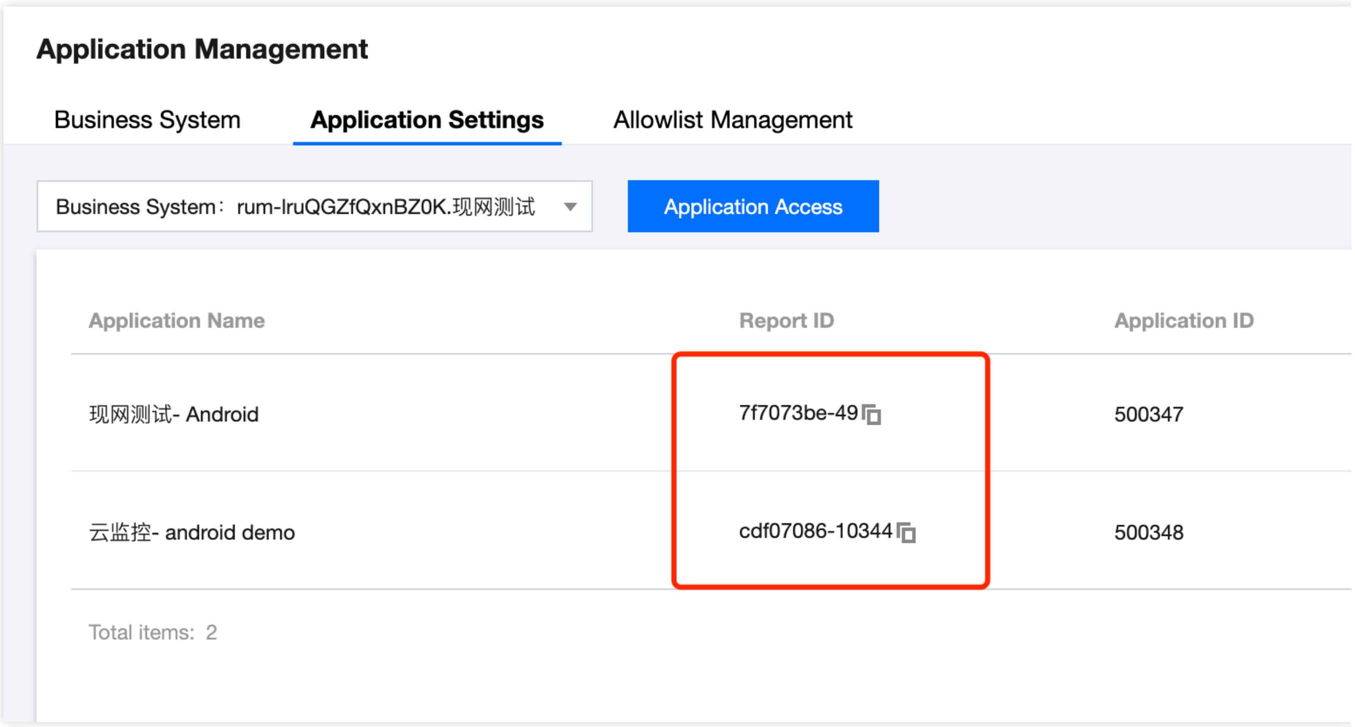
```
<!--Required for information reporting-->
<uses-permission android:name="android.permission.INTERNET" />
<!--Required for information collection-->
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
```

2. To avoid confusion with the SDK, add the following configuration in the app's proguard-rules.pro file:

```
-keep class com.tencent.qapmsdk.**{*;}
# If network monitoring is needed, ensure okhttp3 is not obfuscated.
-keep class okhttp3.**{*;}
```

Step 3: Initialize the SDK.

1. Log in to TCOP console, navigate to the **Mobile App Performance Monitoring** page, select Application Management > Application Settings, and then obtain the Appkey (reporting ID).



Application Name	Report ID	Application ID
现网测试- Android	7f7073be-49	500347
云监控- android demo	cdf07086-10344	500348

Total items: 2

2. Copy the code below and modify some of the fields. The following items are mandatory interface settings; for additional interface configurations, refer to the initialization interface analysis (Initializing QAPM in Application is recommended).

```
// Set the mobile phone model and device ID.
// Device identifier required, in the form of any string. deviceId (mandatory).
// The deviceId can be used to enable an allowlist, avoiding data sampling (Except
QAPM.setProperty(QAPM.PropertyKeyDeviceId, "Device identifier");
// The mobile phone model is required (mandatory).
QAPM.setProperty(QAPM.PropertyKeyModel, "Enter mobile phone model");

// Set Application (mandatory).
QAPM.setProperty(QAPM.PropertyKeyAppInstance, getApplication());
// Set AppKey (mandatory, used to distinguish the reporting products. The value is
QAPM.setProperty(QAPM.PropertyKeyAppId, "YourAppKey");
```

```
// Set the product version, which is used for background retrieval field (mandatory
QAPM.setProperty(QAPM.PropertyKeyAppVersion, "YourApp Version");
// Set the UUID to pull the obfuscated stack's mapping (mandatory. If QAPM Symbol T
// If the qapmplugin plugin is used, this variable will be generated at build time,
QAPM.setProperty(QAPM.PropertyKeySymbolId, BuildConfig.QAPM_UUID);
// Set the user ID, in the form of any string, which is used for background retriev
// The userId can be used to enable an allowlist, avoiding data sampling (except fo
QAPM.setProperty(QAPM.PropertyKeyUserId, "123456");
// Set the Log level (optional). For the production environment version, set it to
QAPM.setProperty(QAPM.PropertyKeyLogLevel, QAPM.LevelInfo);
// Set the QAPM external network reporting domain (required). Chinese Mainland: htt
QAPM.setProperty(QAPM.PropertyKeyHost,"https://app.rumt-zh.com");
QAPM.setProperty(QAPM.PropertyKeyHost,"https://app.rumt-sg.com");
// Enable QAPM.
QAPM.beginScene(QAPM.SCENE_ALL, QAPM.ModeStable);
```

Note:

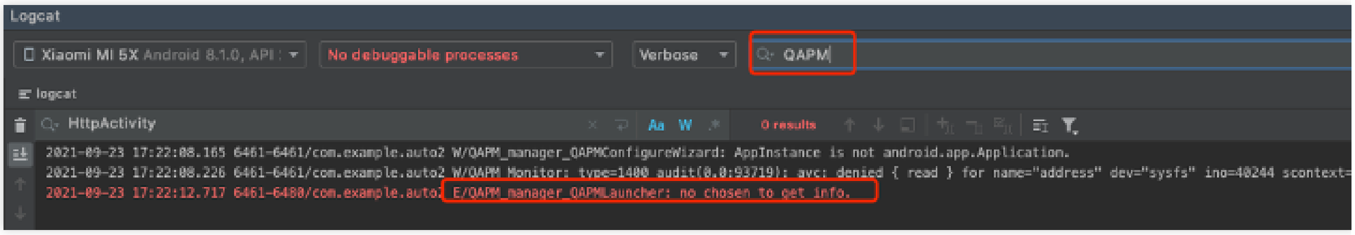
The AppKey can be obtained from the Mobile App Performance Monitoring > Application Management > Application Settings page as in Step 3-Step 1.

Crash and start data are reported in full, while other data is sampled due to its large volume, at a rate of 0.1% (one in a thousand). To report all data, an allowlist can be enabled, and the app will change the sampling rate at the next start. The preset userId or deviceId can be added to the allowlist through the Application Management page to enable the allowlist.

Multiple processes need to initialize QAPM, separately.

Step 4: Access verification.

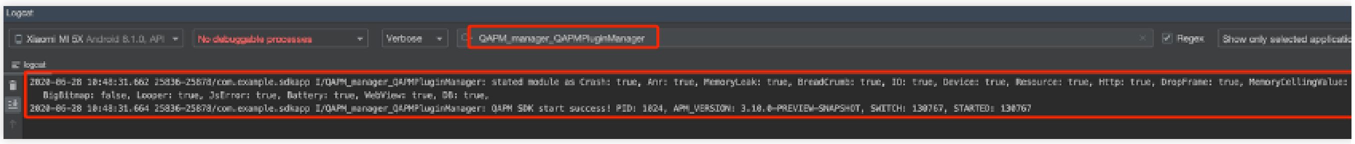
1. If the following log is printed, it indicates that the user has not been sampled, and the sampling rate needs to be reset:



```
2021-09-23 17:22:08.165 6461-6461/com.example.auto2 W/QAPM_manager_QAPMConfigureWizard: AppInstance is not android.app.Application.
2021-09-23 17:22:08.226 6461-6461/com.example.auto2 W/QAPM_Monitor: type=1400 audit(0.0:93719): avc: denied { read } for name="address" dev="sysfs" ino=40244 scontext=
2021-09-23 17:22:12.717 6461-6480/com.example.auto2 E/QAPM_manager_QAPMLauncher: no chosen to get info.
```

See TAG: QAPM_manager_QAPMLauncher

2. If the following log is printed, it indicates that the initial access succeeded. You can verify data reporting and try to enable the advanced feature:


Logcat
Xiaomi MI 5X Android 8.1.0, API
No debuggable processes
Verbose
QAPM_manager_QAPMPluginManager
Regex
Show only selected applicatio
logcat

```
2020-06-28 10:48:31.662 25836-25878/com.example.sdkapp I/QAPM_manager_QAPMPluginManager: stated module as Crash: true, Anr: true, MemoryLeak: true, BreadCrumb: true, IO: true, Device: true, Resource: true, Http: true, DropFrame: true, MemoryCellingValue:
    BigBitmap: false, Looper: true, JsError: true, Battery: true, WebView: true, DB: true,
2020-06-28 10:48:31.664 25836-25878/com.example.sdkapp I/QAPM_manager_QAPMPluginManager: QAPM SDK start success! PID: 1024, APM_VERSION: 3.10.0-PREVIEW-SNAPSHOT, SWITCH: 130767, STARTED: 130767
```

See TAG: QAPM_manager_QAPMPluginManager

Initialized Interface Analysis

API Name	Parameter	Parameter Description	Notes
public static QAPM setProperty(int key, Stringvalue) **Functionality**: Set QAPM parameters.	key	**Required**. The Key that needs to be set.	-
	QAPM.PropertyKeyLogLevel	**Optional**. Enable log level. (It is recommended to use QAPM.LevelDebug for Debug versions and QAPM.LevelWarn for release versions).	
	QAPM.PropertyNeedTranslate	**Optional**. Whether stack translation is needed, which by default is required. If the apk is not obfuscated, then pass in 'false'. Otherwise the frontend may display everything as 'unTranslated'.	
public static boolean beginScene(String sceneName, int mode) **Functionality**: Enable monitoring.	sceneName	**Required**. Scene name.	Use the OR operation to enable the performance module from Definition, such as enabling Crash and Anr: beginScene(“Crash&ANR”, QAPM.ModeCrash| QAPM.ModeANR)
	mode	**Required**. The feature to be enabled.	
	QAPM.ModeStable	**Optional**. Enable all features	

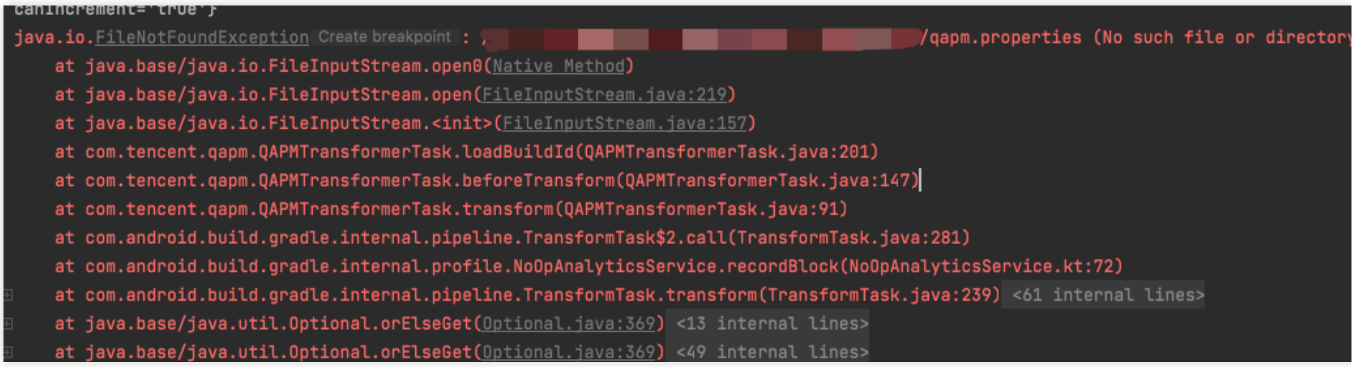
		(Recommended for external releases. Includes interval performance, crash, ANR, WebView page load, JsError, and network).	
	QAPM.ModeWebView	**Optional**. Enable WebView page load monitoring.	
	QAPM.ModeJsError	**Optional**. Enable WebView JS exception monitoring.	
	QAPM.ModeHTTPInWeb	**Optional**. Enable WebView network monitoring.	
	QAPM.ModeHTTP	**Optional**. Enable network monitoring.	
public static boolean endScene(String sceneName, long mode) **Functionality**: End monitoring (Only effective for frame drops and interval performance collection).	sceneName	**Required**. The name of the scene to be turned off (Required to correspond to the beginScene).	
	QAPM.ModeDropFrame	**Optional**. Turn off frame drop monitoring.	-
	QAPM.ModeResource	Optional. Turn off interval performance monitoring.	

Others

Note:

When compiling and packaging the app through the qapm plugin, the app needs a UUID as the Build ID. If there is a qapm.properties file in the project directory, and the value of the qapm_uuid property exists, this value will be used as the Build ID; otherwise, the plugin will randomly generate a Build ID.

qapm-plugin Version 2.39 and earlier will report an IO Error during the app compiling process:
`java.io.FileNotFoundException, qapm.properties (No such file or directory)`.



```
canIncrement='true'}
java.io.FileNotFoundException Create breakpoint : [redacted]/qapm.properties (No such file or directory
    at java.base/java.io.FileInputStream.open0(Native Method)
    at java.base/java.io.FileInputStream.open(FileInputStream.java:219)
    at java.base/java.io.FileInputStream.<init>(FileInputStream.java:157)
    at com.tencent.qapm.QAPMTransformerTask.loadBuildId(QAPMTransformerTask.java:201)
    at com.tencent.qapm.QAPMTransformerTask.beforeTransform(QAPMTransformerTask.java:147)
    at com.tencent.qapm.QAPMTransformerTask.transform(QAPMTransformerTask.java:91)
    at com.android.build.gradle.internal.pipeline.TransformTask$2.call(TransformTask.java:281)
    at com.android.build.gradle.internal.profile.NoOpAnalyticsService.recordBlock(NoOpAnalyticsService.kt:72)
    at com.android.build.gradle.internal.pipeline.TransformTask.transform(TransformTask.java:239) <61 internal lines>
    at java.base/java.util.Optional.orElseGet(Optional.java:369) <13 internal lines>
    at java.base/java.util.Optional.orElseGet(Optional.java:369) <49 internal lines>
```

This error only occurs during compilation and does not affect the running of the app.

Feature Configuration

Network Monitoring

Last updated：2024-05-14 12:36:06

Enabling Feature

Network monitoring requires the use of the qapm-plugin for instrumentation and is by default inserted at various entry and exit points of the network layer.

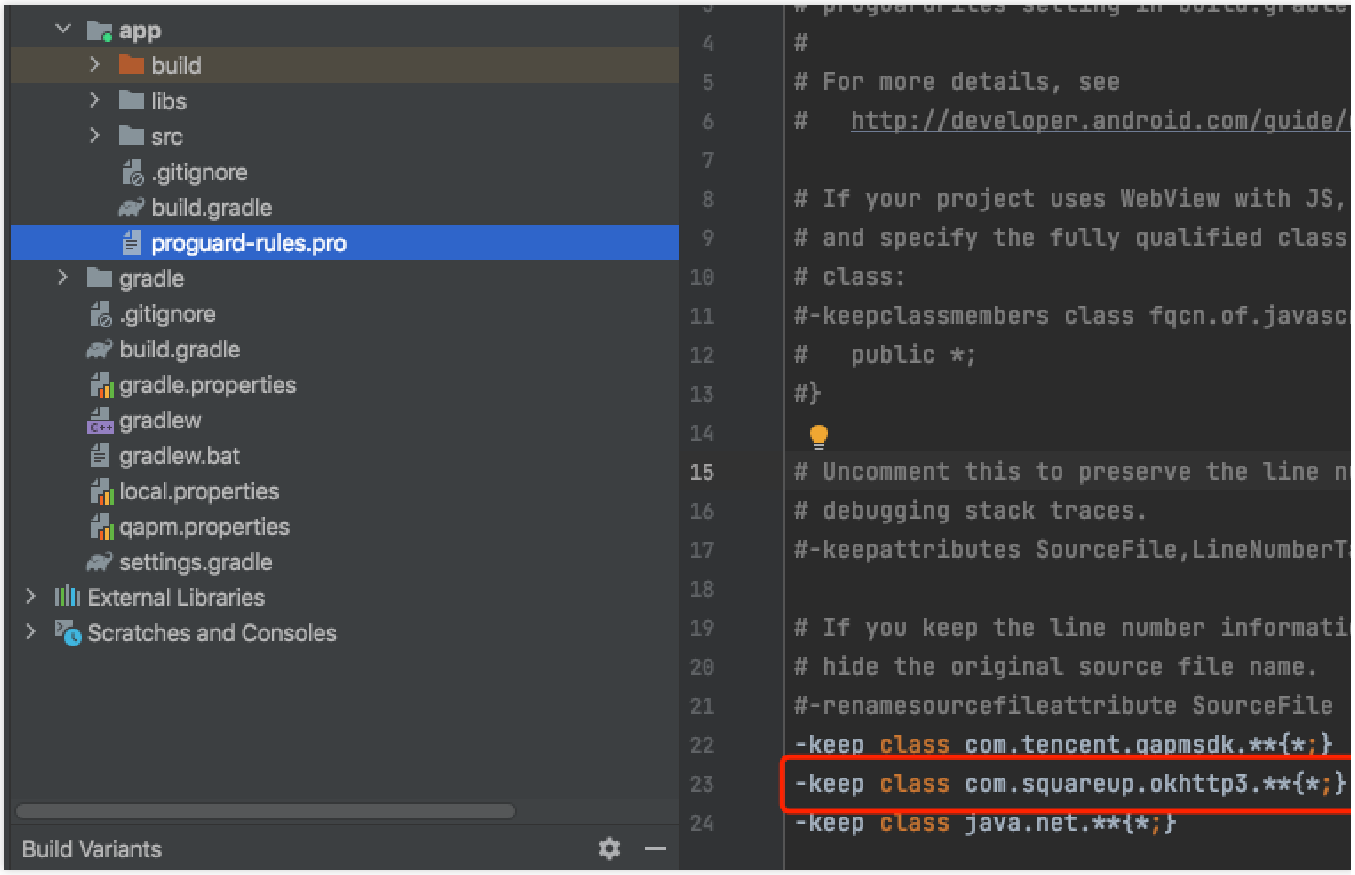
Prerequisites

A user allowlist or device allowlist, which is used to initialize the SDK, has been added through the Mobile App Performance Monitoring > Application Management > Allowlist Management page.
The qapm-plugin has been configured in the app-level build.gradle. See Integration and Initialization.
Currently, only okhttp3 monitoring is supported. The okhttp3 also requires okio version 1.14.0 or later.

Configuration Process

Add obfuscation rules in the proguard-rules.pro file in the app directory to prevent okttp3 code from being obfuscated.

```
-keep class com.squareup.okhttp3.**{*;}
```

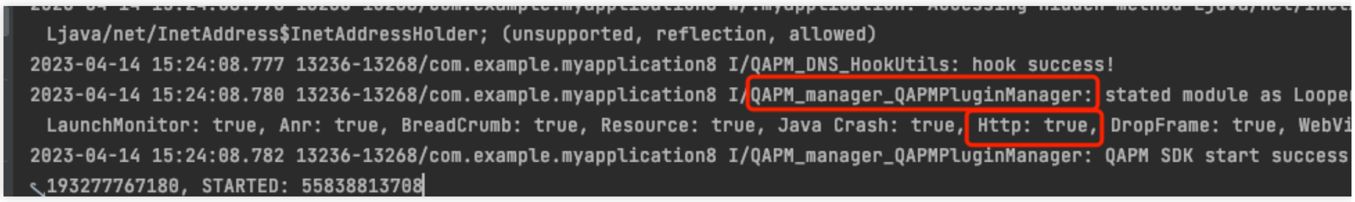


```
#
# For more details, see
#   http://developer.android.com/guide/
 
# If your project uses WebView with JS,
# and specify the fully qualified class
# class:
#-keepclassmembers class fqcn.of.javasc
#   public *;
#}

# Uncomment this to preserve the line n
# debugging stack traces.
#-keepattributes SourceFile,LineNumberT

# If you keep the line number informati
# hide the original source file name.
#-renamesourcefileattribute SourceFile
-keep class com.tencent.qapmsdk.**{*;}
-keep class com.squareup.okhttp3.**{*;}
-keep class java.net.**{*;}
```

Verifying Whether the Feature Is Working Properly

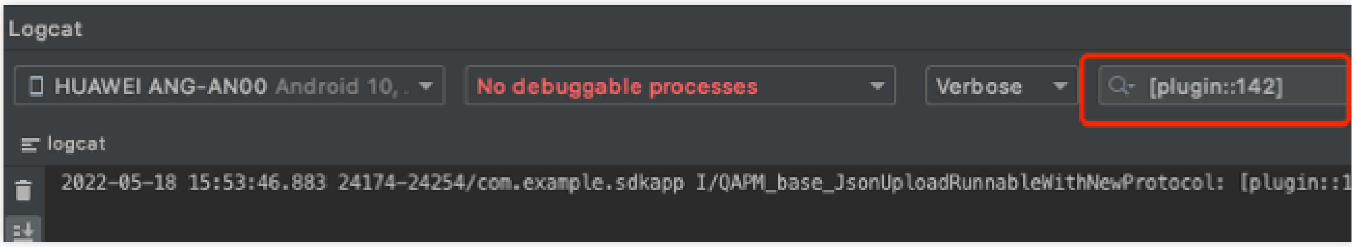
Retrieval tag:QAPM_manager_QAPMPluginManager

The log message that appears one minute after each network request indicates successful reporting of network data:



```
  Ljava/net/InetAddress$InetAddressHolder; (unsupported, reflection, allowed)
2023-04-14 15:24:08.777 13236-13268/com.example.myapplication8 I/QAPM_DNS_HookUtils: hook success!
2023-04-14 15:24:08.780 13236-13268/com.example.myapplication8 I/QAPM_manager_QAPMPluginManager: stated module as Loope
  LaunchMonitor: true, Anr: true, BreadCrumb: true, Resource: true, Java Crash: true, Http: true, DropFrame: true, WebVi
2023-04-14 15:24:08.782 13236-13268/com.example.myapplication8 I/QAPM_manager_QAPMPluginManager: QAPM SDK start success
  193277767180, STARTED: 55838813708
```

Retrieval tag: [plugin::142]



```
2022-05-18 15:53:46.883 24174-24254/com.example.sdkapp I/QAPM_base_JsonUploadRunnableWithNewProtocol: [plugin::1
```

Note:

It requires the use of the qapm-plugin for instrumentation. Otherwise, it will not work.

The SDK is only responsible for capturing information related to network requests. The backend analyzes issue data, such as slow requests (with the request time being greater than xxs) and network errors (with the request response code being greater than 400).

Data can be viewed in Mobile Performance Monitoring > Network > Slow Requests and Error Requests List.

If the correct allowlist is not configured, the SDK will not enable network monitoring.

WebView, JsError, and Web Network Monitoring

Last updated：2024-05-14 12:36:06

Enabling Feature

Initialization requires enabling WebView, JsError, and Web network monitoring. Below is how to enable these three features based on Stable. The code is as follows:

```
QAPM.beginScene(QAPM.SCENE_ALL, QAPM.ModeStable | QAPM.ModeWebView | QAPM.ModeJsErr
```

In addition, the following code needs to be configured:

WebView Monitoring requires enabling interaction with JavaScript. Call the following code during WebView initialization to enable:

```
WebSettings webSetting = webView.getSettings();
webSetting.setJavaScriptEnabled(true);
```

After WebView initialization, add a call interface channel for Java and JS. The purpose is to allow the JS layer to obtain some configuration information from the Java layer:

```
webView.addJavascriptInterface(QAPMJavaScriptBridge.getInstance(),"QAPMAndroidJsBri
```

Add the following method in the WebView's shouldInterceptRequest code to intercept web-sdk and replace it with local SDK resources. **Make sure to call the following code at the earliest position in this callback**.

If it is x5, use the following code:

```
@Overridepublic
public WebResourceResponse shouldInterceptRequest(WebView webView, String s) {???
    Object response = QAPMJavaScriptBridge.getInstance().shouldInterceptRequestWith
    if (response != null) {???????
        return (WebResourceResponse)response;???
     ??? }
     return super.shouldInterceptRequest(webView, s);
     ??? }
```

If it is Native WebView, use the following code:

```
@Overridepublic
public WebResourceResponse shouldInterceptRequest(WebView webView, String s) {???
    WebResourceResponse response = QAPMJavaScriptBridge.getInstance().shouldInterce
    if (response != null) {???????
        return response;???
    ??? }
    return super.shouldInterceptRequest(webView, s);
  ??? }
```

Add the following method in the WebView's onPageFinished code to inject JS script:

```
webView.setWebViewClient(new WebViewClient(){
    ??? @Override
    ??? public void onPageFinished(WebView view, String url) {
        ??????? super.onPageFinished(view, url);
        ??????? QAPMJavaScriptBridge.getInstance().initFileJS(view);
    ??? }
});
```

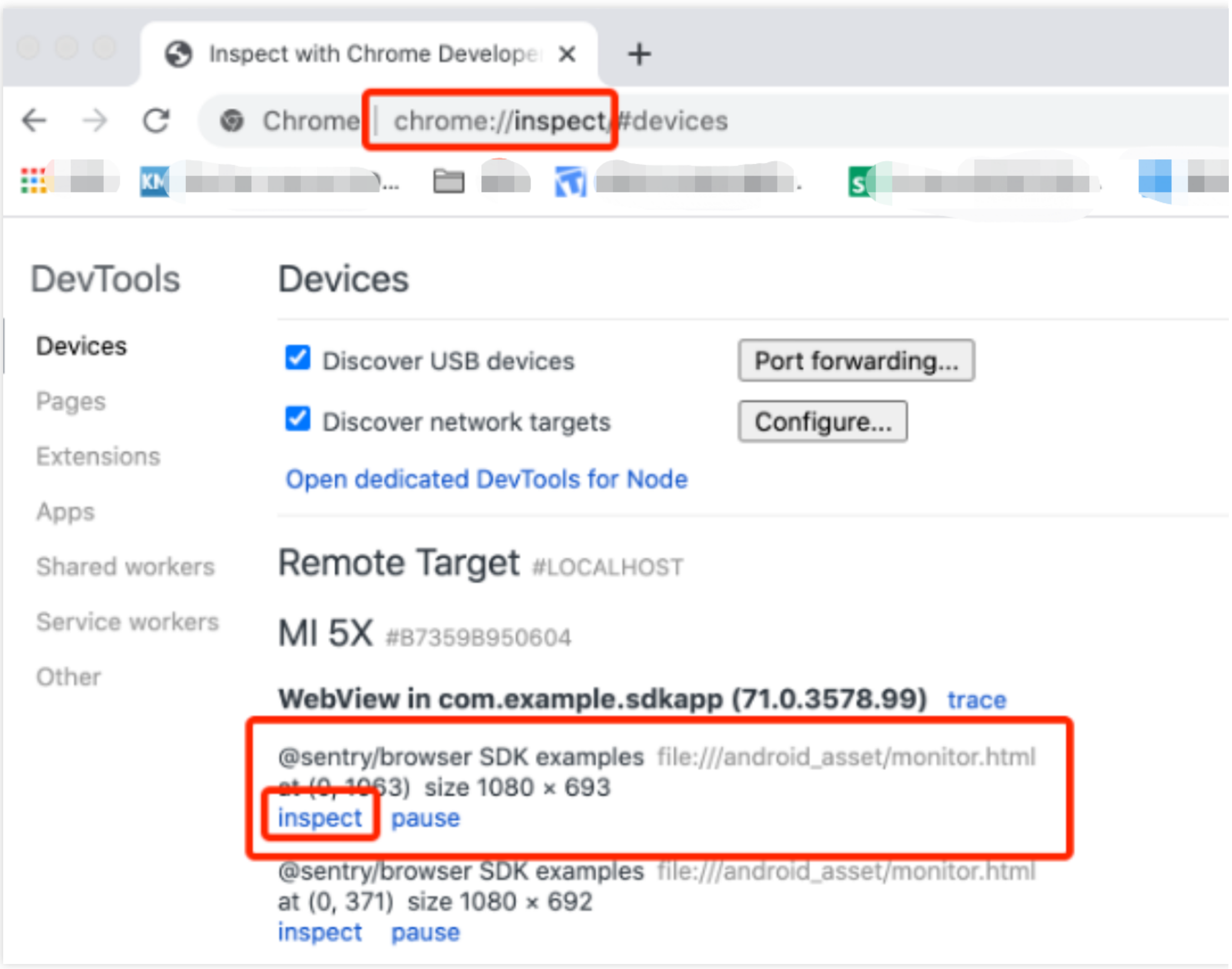
Verifying Whether the Feature Is Working Properly

Native WebView, JsError monitoring:

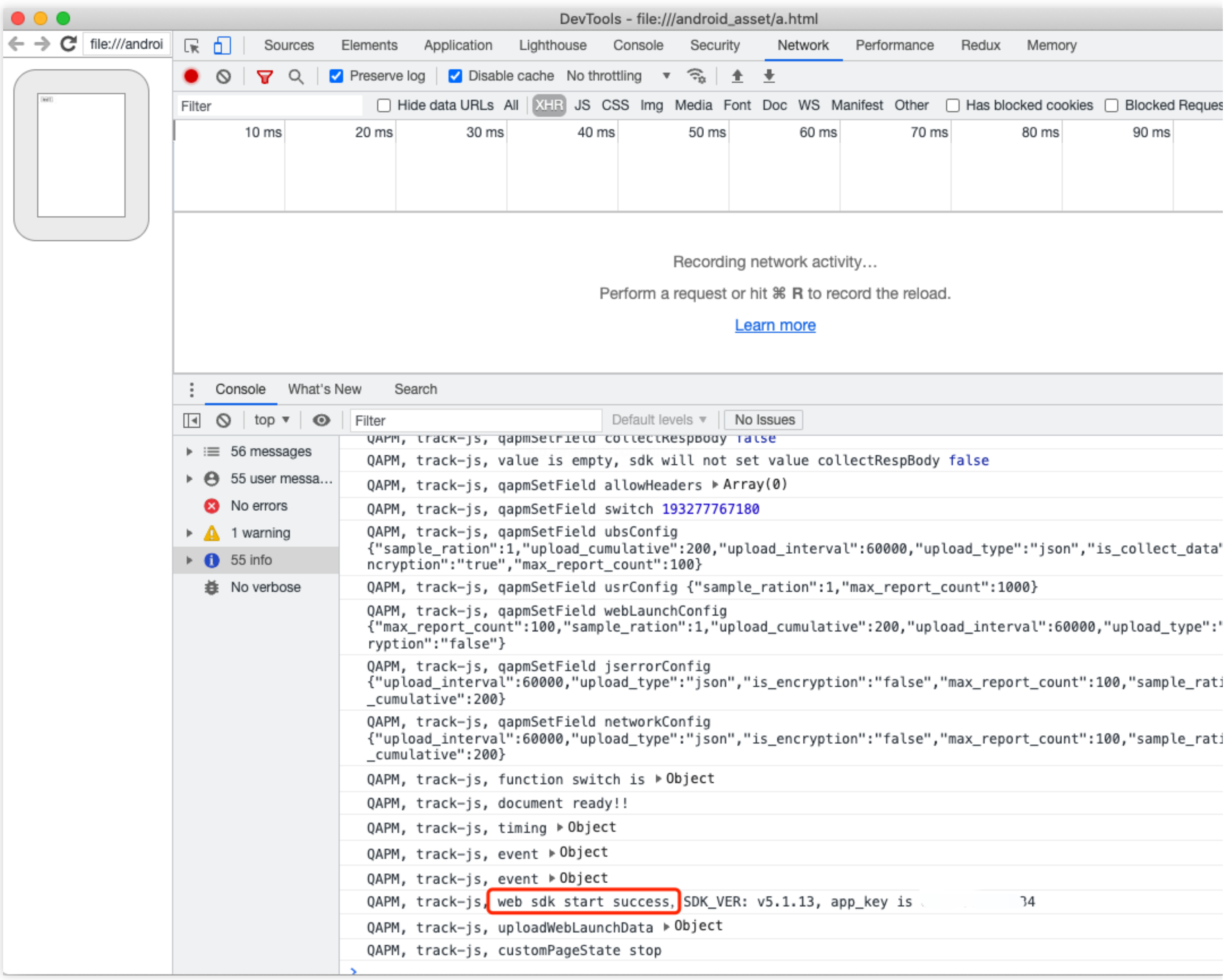
1. Include the following code in the code (for remote debugging purposes).

```
WebView.setWebContentsDebuggingEnabled(true);
```

2. Open Google Chrome, and enter `chrome://inspect` in the address bar. In the devices that appear, click **inspect**.


Inspect with Chrome Develope
Chrome | chrome://inspect/#devices
DevTools
Devices
Pages
Extensions
Apps
Shared workers
Service workers
Other
Devices
Discover USB devices
Port forwarding...
Discover network targets
Configure...
Open dedicated DevTools for Node
Remote Target #LOCALHOST
MI 5X #B7359B950604
WebView in com.example.sdkapp (71.0.3578.99) trace
@sentry/browser SDK examples file:///android_asset/monitor.html
at (0, 1063) size 1080 × 693
inspect pause
@sentry/browser SDK examples file:///android_asset/monitor.html
at (0, 371) size 1080 × 692
inspect pause

3. Find the Console module query log. If `web start success, vxxx` is displayed, it indicates the WebSDK inject succeeded.

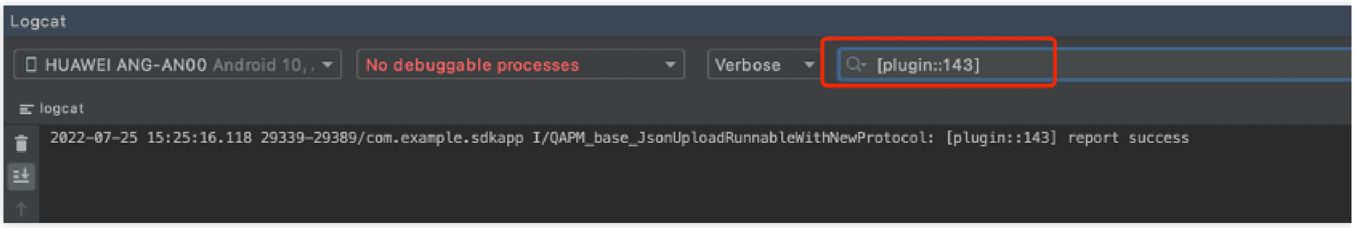


```
QAPM, track-js, qapmSetField collectRespBody false
QAPM, track-js, value is empty, sdk will not set value collectRespBody false
QAPM, track-js, qapmSetField allowHeaders ▸Array(0)
QAPM, track-js, qapmSetField switch 193277767180
QAPM, track-js, qapmSetField ubsConfig
{"sample_ration":1,"upload_cumulative":200,"upload_interval":60000,"upload_type":"json","is_collect_data"
ncryption":"true","max_report_count":100}
QAPM, track-js, qapmSetField usrConfig {"sample_ration":1,"max_report_count":1000}
QAPM, track-js, qapmSetField webLaunchConfig
{"max_report_count":100,"sample_ration":1,"upload_cumulative":200,"upload_interval":60000,"upload_type":"
ryption":"false"}
QAPM, track-js, qapmSetField jserrorConfig
{"upload_interval":60000,"upload_type":"json","is_encryption":"false","max_report_count":100,"sample_rati
_cumulative":200}
QAPM, track-js, qapmSetField networkConfig
{"upload_interval":60000,"upload_type":"json","is_encryption":"false","max_report_count":100,"sample_rati
_cumulative":200}
QAPM, track-js, function switch is ▸Object
QAPM, track-js, document ready!!
QAPM, track-js, timing ▸Object
QAPM, track-js, event ▸Object
QAPM, track-js, event ▸Object
QAPM, track-js, web sdk start success, SDK_VER: v5.1.13, app_key is [redacted]34
QAPM, track-js, uploadWebLaunchData ▸Object
QAPM, track-js, customPageState stop
```

4. Check whether all features are reporting normally. Consider JsError reporting as an example, as follows:

Retrieval tag: [plugin::143].

The occurrence of a JsError, such as the following log message, indicates successful reporting of JsError data.



```
2022-07-25 15:25:16.118 29339-29389/com.example.sdkapp I/QAPM_base_JsonUploadRunnableWithNewProtocol: [plugin::143] report success
```

The other retrieval tags are as follows:

Page load: plugin::141 (reported immediately after each Web page load is complete).

Network request: plugin::154 (reported when there are network errors and slow requests).

Note:

1. To check whether WebView monitoring is normal, examine through browsers like Chrome for debugging.

2. Page load is reported in the Issue Case Details only if the page load duration exceeds 3.5s.
3. Network requests are reported in cases of network errors and slow networks.

Crash and ANR Monitoring

Last updated：2024-05-14 12:36:06

Enabling Feature

Initialization requires enabling Crash and ANR monitoring, which by default monitors Crash and ANR information.

```
// ModeStable mode by default includes Crash and ANR monitoring.
```

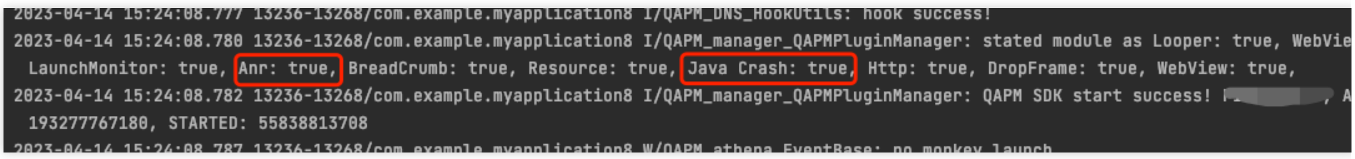
```
QAPM.beginScene(QAPM.SCENE_ALL, QAPM.ModeStable);
```

QAPM provides APIs for uploading custom log files in case of crashes or ANRs, if necessary. An example is as follows:

```
QAPM.setProperty(QAPM.PropertyExtraDataListener, new IExtraDataListener() {
    // This callback is executed when an ANR occurs.
    @Override
    public List<String> onAnrExtraFileHandler() {
        List<String> files = new ArrayList<>();
        File[] fileArray = new File("xxxx").listFiles();//Enter the folder name at
        for (File file : fileArray) {
            files.add(file.getAbsolutePath());
        }
        return files;
    }
    // This callback is executed when a crash occurs.
    @Override
    public List<String> onCrashExtraFileHandler() {
        List<String> files = new ArrayList<>();
        File[] fileArray = new File("xxxx").listFiles();//Enter the folder name at
        for (File file : fileArray) {
            files.add(file.getAbsolutePath());
        }
        return files;
    }
});
```

Verifying Whether the Feature Is Working Properly

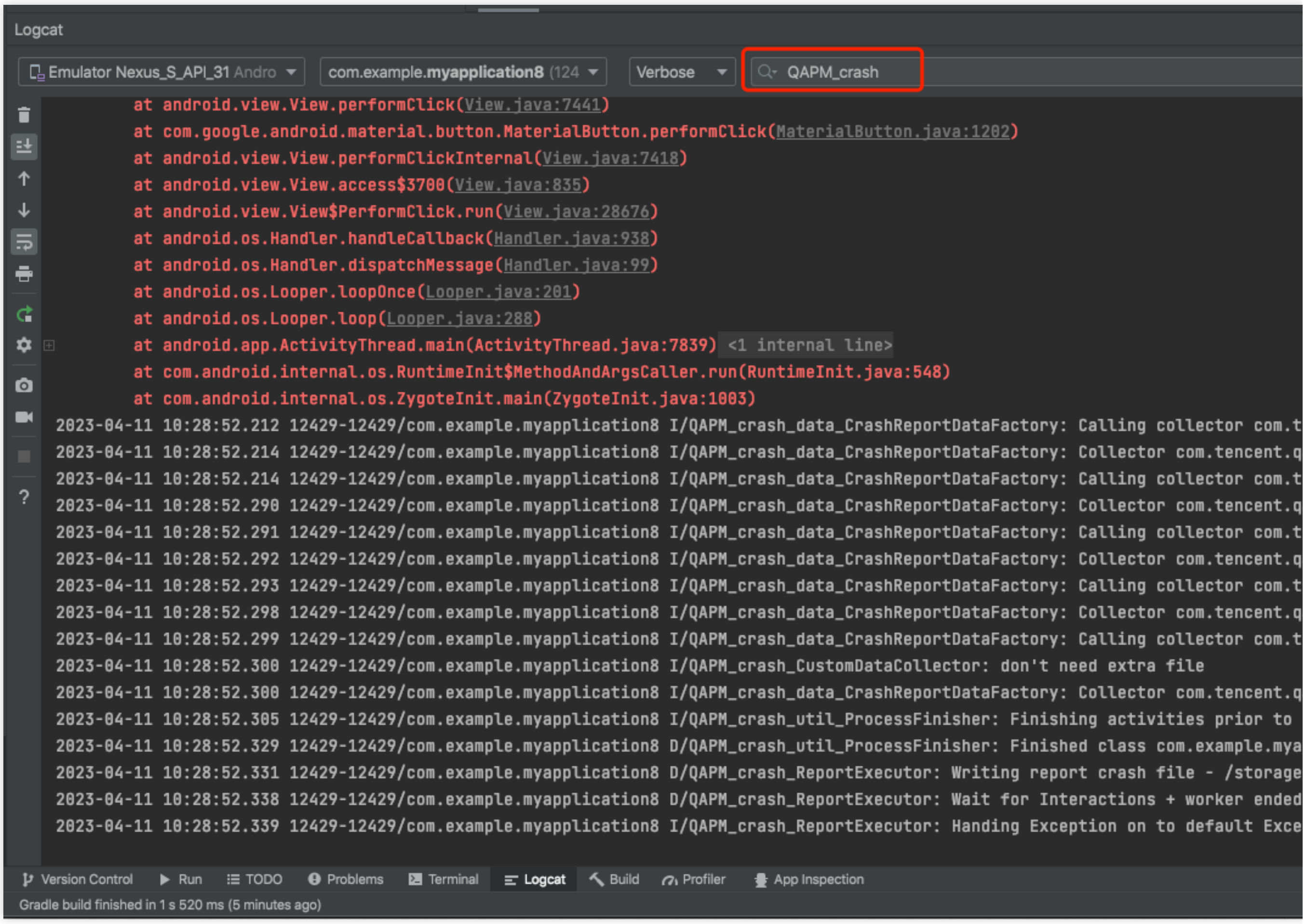
Retrieval tag: QAPM_manager_QAPMPluginManager



```
2023-04-14 15:24:08.777 13236-13268/com.example.myapplication8 I/QAPM_DNS_HookUtils: hook success!
2023-04-14 15:24:08.780 13236-13268/com.example.myapplication8 I/QAPM_manager_QAPMPluginManager: stated module as Looper: true, WebVie
 LaunchMonitor: true, Anr: true, BreadCrumb: true, Resource: true, Java Crash: true, Http: true, DropFrame: true, WebView: true,
2023-04-14 15:24:08.782 13236-13268/com.example.myapplication8 I/QAPM_manager_QAPMPluginManager: QAPM SDK start success! [redacted], A
 193277767180, STARTED: 55838813708
2023-04-14 15:24:08.787 13236-13268/com.example.myapplication8 W/QAPM_athena_EventBase: no monkey launch
```

Retrieval tag: QAPM_crash

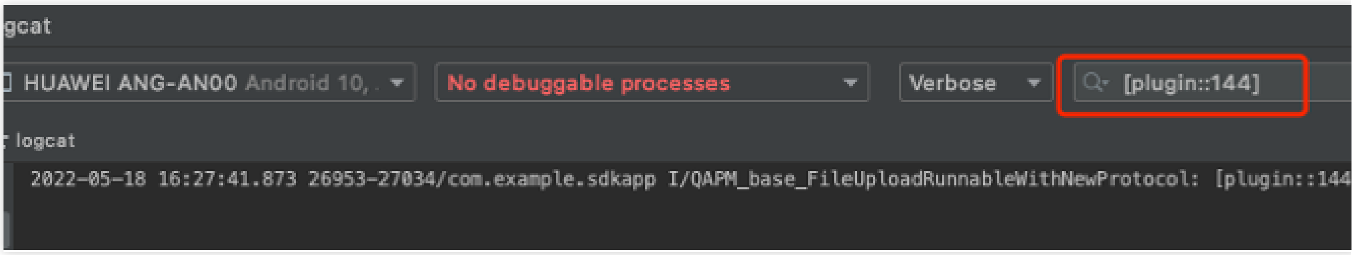
The following log message in case of crashes or ANRs indicates that QAPM has collected this exception:


Logcat
Emulator Nexus_S_API_31 Andro
com.example.myapplication8 (124
Verbose
QAPM_crash
Version Control
Run
TODO
Problems
Terminal
Logcat
Build
Profiler
App Inspection
Gradle build finished in 1 s 520 ms (5 minutes ago)

```
        at android.view.View.performClick(View.java:7441)
        at com.google.android.material.button.MaterialButton.performClick(MaterialButton.java:1202)
        at android.view.View.performClickInternal(View.java:7418)
        at android.view.View.access$3700(View.java:835)
        at android.view.View$PerformClick.run(View.java:28676)
        at android.os.Handler.handleCallback(Handler.java:938)
        at android.os.Handler.dispatchMessage(Handler.java:99)
        at android.os.Looper.loopOnce(Looper.java:201)
        at android.os.Looper.loop(Looper.java:288)
        at android.app.ActivityThread.main(ActivityThread.java:7839) <1 internal line>
        at com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:548)
        at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:1003)
2023-04-11 10:28:52.212 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Calling collector com.t
2023-04-11 10:28:52.214 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Collector com.tencent.q
2023-04-11 10:28:52.214 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Calling collector com.t
2023-04-11 10:28:52.290 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Collector com.tencent.q
2023-04-11 10:28:52.291 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Calling collector com.t
2023-04-11 10:28:52.292 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Collector com.tencent.q
2023-04-11 10:28:52.293 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Calling collector com.t
2023-04-11 10:28:52.298 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Collector com.tencent.q
2023-04-11 10:28:52.299 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Calling collector com.t
2023-04-11 10:28:52.300 12429-12429/com.example.myapplication8 I/QAPM_crash_CustomDataCollector: don't need extra file
2023-04-11 10:28:52.300 12429-12429/com.example.myapplication8 I/QAPM_crash_data_CrashReportDataFactory: Collector com.tencent.q
2023-04-11 10:28:52.305 12429-12429/com.example.myapplication8 I/QAPM_crash_util_ProcessFinisher: Finishing activities prior to
2023-04-11 10:28:52.329 12429-12429/com.example.myapplication8 D/QAPM_crash_util_ProcessFinisher: Finished class com.example.mya
2023-04-11 10:28:52.331 12429-12429/com.example.myapplication8 D/QAPM_crash_ReportExecutor: Writing report crash file - /storage
2023-04-11 10:28:52.338 12429-12429/com.example.myapplication8 D/QAPM_crash_ReportExecutor: Wait for Interactions + worker ended
2023-04-11 10:28:52.339 12429-12429/com.example.myapplication8 I/QAPM_crash_ReportExecutor: Handing Exception on to default Exce
```

Retrieval tag: plugin::144

The following log message indicates that QAPM has successfully reported this exception. An example is as follows:



```
2022-05-18 16:27:41.873 26953-27034/com.example.sdkapp I/QAPM_base_FileUploadRunnableWithNewProtocol: [plugin::144
```

The other crash retrieval tags are as follows:

ANR: [plugin::140].

NativeCrash: [plugin::146].

Note:

To avoid lagging, keep the logic in the interface callbacks as simple and straightforward as possible.

Uploaded files must be less than 20 MB in size. Files larger than the limit will not be uploaded. Select helpful log files.

Crash events can be viewed on the Mobile Monitoring Crash page, and ANR rates can be viewed in the Overview page.

Lag and Frame Rate Monitoring

Last updated：2024-05-14 12:36:06

Prerequisites

Integration and Initialization has been Integration and Initialization.

Feature Configuration

Enabling Monitoring

Initialization requires enabling lag monitoring. Lag doesn't need instrumentation, while frame loss rate requires additional instrumentation. It is recommended to add tracking on scrolling lists, such as (ListView, GridView, and RecyclerView).

Frame Loss Rate Instrumentation

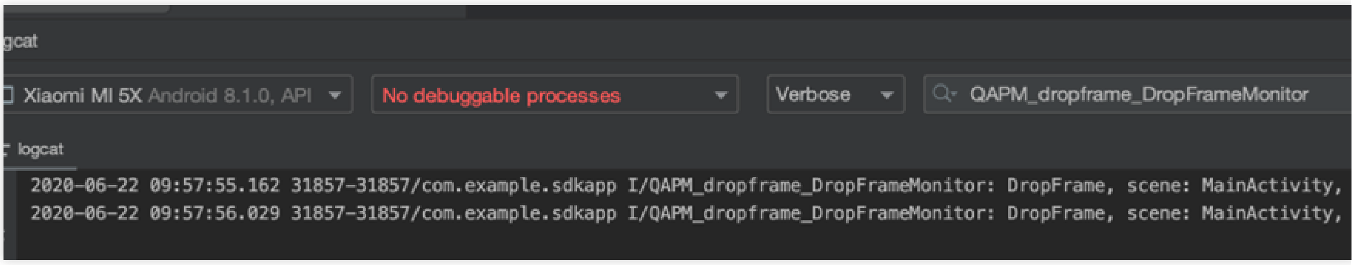
Call QAPM.beginScene("xxx scrolling", QAPM.ModeDropFrame) before each scroll.
Call QAPM.endScene("xxx scrolling", QAPM.ModeDropFrame) after a scroll ends.
This can generally be achieved by overriding the scrolling component's onScrollStateChanged method, as shown below:

```
@Override
public void onScrollStateChanged(AbsListView view, int scrollState) {
    if (scrollState == AbsListView.OnScrollListener.SCROLL_STATE_IDLE) {
        QAPM.endScene("xxx scrolling", QAPM.ModeDropFrame); //xxx scrolling name ca
    } else {
        QAPM.beginScene("xxx scrolling", QAPM.ModeDropFrame);//xxx scrolling name c
    }
```

Verifying Whether the Feature Is Working Properly

Retrieval tag:QAPM_dropframe_DropFrameMonitor

After a scroll ends (endScene calling), the following log message indicates that the frame loss rate data has been stored in the local database:



gcat

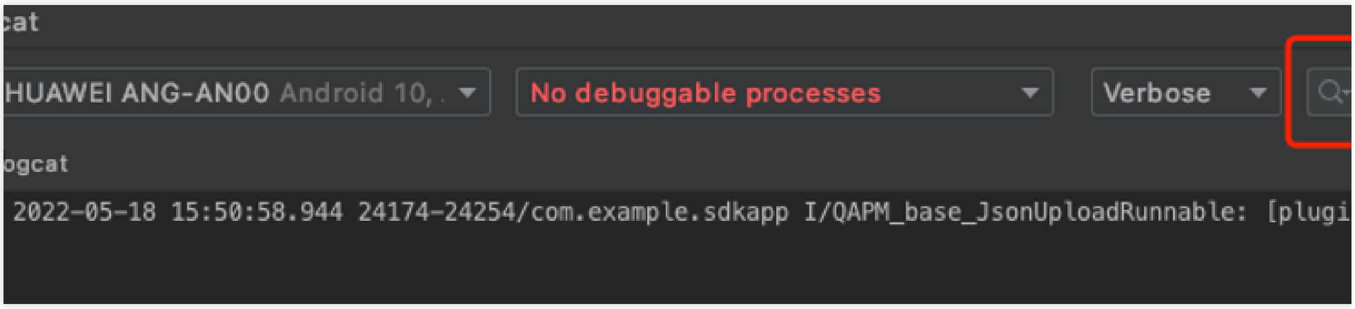
Xiaomi MI 5X Android 8.1.0, API | No debuggable processes | Verbose | QAPM_dropframe_DropFrameMonitor

logcat

```
2020-06-22 09:57:55.162 31857-31857/com.example.sdkapp I/QAPM_dropframe_DropFrameMonitor: DropFrame, scene: MainActivity,
2020-06-22 09:57:56.029 31857-31857/com.example.sdkapp I/QAPM_dropframe_DropFrameMonitor: DropFrame, scene: MainActivity,
```

Retrieval tag: [plugin::101]

The following log message indicates successful reporting of frame loss data that is stored in the app's local database.



cat

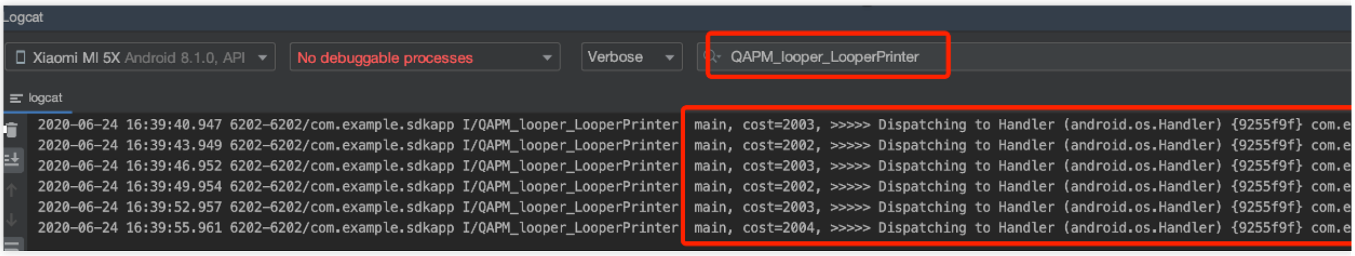
HUAWEI ANG-AN00 Android 10, | No debuggable processes | Verbose

ogcat

```
2022-05-18 15:50:58.944 24174-24254/com.example.sdkapp I/QAPM_base_JsonUploadRunnable: [plugi
```

Retrieval tag: QAPM_looper_LooperPrinter

The following log message indicates that Lag Monitoring is functioning properly:



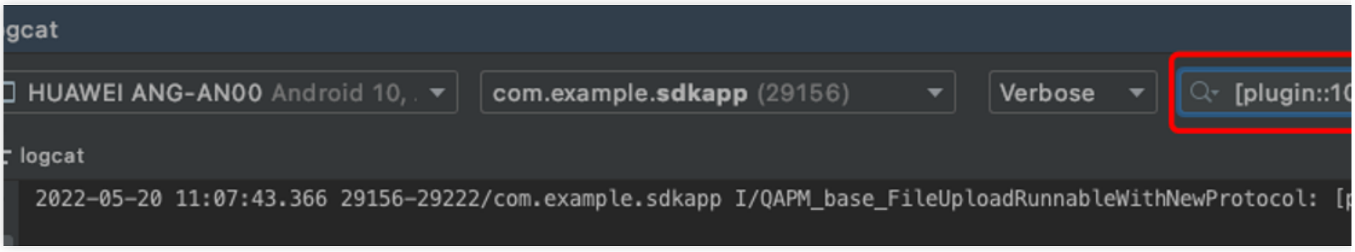
Logcat

Xiaomi MI 5X Android 8.1.0, API | No debuggable processes | Verbose | QAPM_looper_LooperPrinter

logcat

```
2020-06-24 16:39:40.947 6202-6202/com.example.sdkapp I/QAPM_looper_LooperPrinter main, cost=2003, >>>>> Dispatching to Handler (android.os.Handler) {9255f9f} com.e
2020-06-24 16:39:43.949 6202-6202/com.example.sdkapp I/QAPM_looper_LooperPrinter main, cost=2002, >>>>> Dispatching to Handler (android.os.Handler) {9255f9f} com.e
2020-06-24 16:39:46.952 6202-6202/com.example.sdkapp I/QAPM_looper_LooperPrinter main, cost=2003, >>>>> Dispatching to Handler (android.os.Handler) {9255f9f} com.e
2020-06-24 16:39:49.954 6202-6202/com.example.sdkapp I/QAPM_looper_LooperPrinter main, cost=2002, >>>>> Dispatching to Handler (android.os.Handler) {9255f9f} com.e
2020-06-24 16:39:52.957 6202-6202/com.example.sdkapp I/QAPM_looper_LooperPrinter main, cost=2003, >>>>> Dispatching to Handler (android.os.Handler) {9255f9f} com.e
2020-06-24 16:39:55.961 6202-6202/com.example.sdkapp I/QAPM_looper_LooperPrinter main, cost=2004, >>>>> Dispatching to Handler (android.os.Handler) {9255f9f} com.e
```

The following log message indicates that Lag Reporting is functioning properly:



```
2022-05-20 11:07:43.366 29156-29222/com.example.sdkapp I/QAPM_base_FileUploadRunnableWithNewProtocol: [p
```

Startup Monitoring

Last updated：2024-05-14 12:36:06

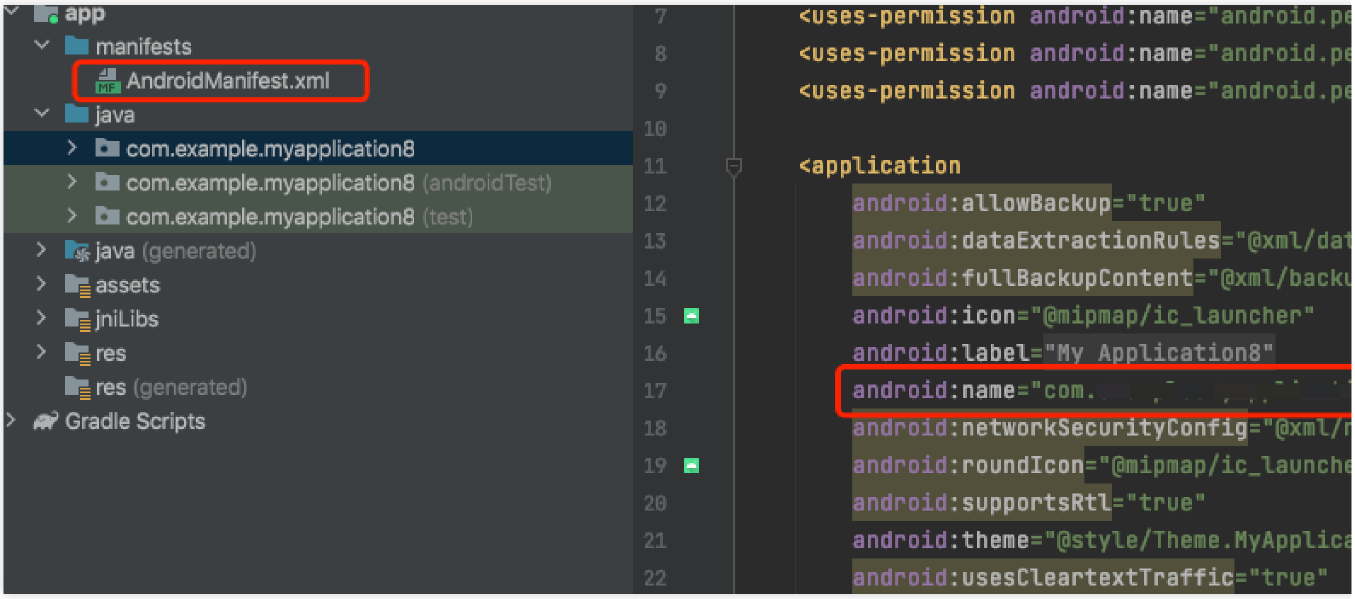
Startup monitoring requires the use of the qapm-plugin plugin for instrumentation during compilation. The default instrumentation points are the various lifecycles of the Application and Activity. In the App SDK, the default startup time is measured from Application's attachBaseContext to the end of onResume of the first Activity.

Prerequisites

The qapm-plugin has been configured in the app-level build.gradle.

Configuration Process

1. Manually add an Application subclass, such as BaseApplication (the name is not restricted, and the subclass does not need to implement any methods or add any attributes).
2. In the AndroidManifest.xml file, add the android:name attribute to the application node, with the value being "package name+Application subclass name".


app
manifests
AndroidManifest.xml
java
com.example.myapplication8
com.example.myapplication8 (androidTest)
com.example.myapplication8 (test)
java (generated)
assets
jniLibs
res
res (generated)
Gradle Scripts

```
    <uses-permission android:name="android.pe
    <uses-permission android:name="android.pe
    <uses-permission android:name="android.pe

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/dat
        android:fullBackupContent="@xml/backu
        android:icon="@mipmap/ic_launcher"
        android:label="My Application8"
        android:name="com.[redacted]
        android:networkSecurityConfig="@xml/n
        android:roundIcon="@mipmap/ic_launche
        android:supportsRtl="true"
        android:theme="@style/Theme.MyApplica
        android:usesCleartextTraffic="true"
```

Additional Tracking

If you want to measure the execution time of certain methods within the startup interval, additional tracking is required, as shown below:
Refer to the code:

```
QAPM.beginScene(StageConstant.QAPM_APPLAUNCH, "Method Name", QAPM.ModeResource);
/**Service logic*/
QAPM.endScene(StageConstant.QAPM_APPLAUNCH, "Method Name", QAPM.ModeResource);
```

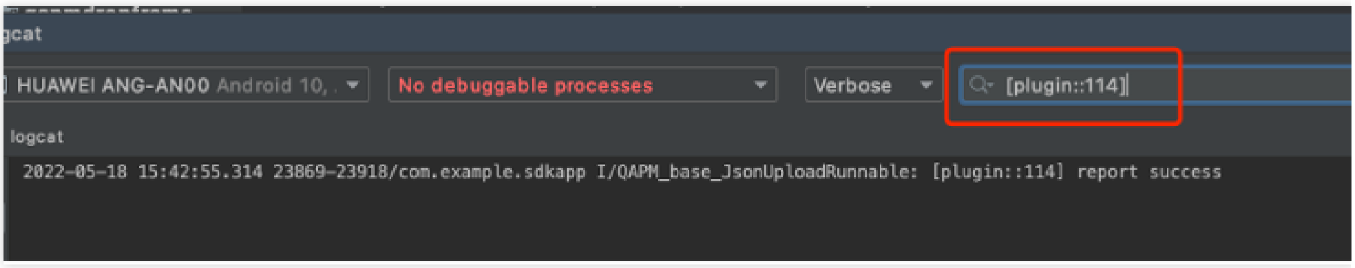
If you want to set your own end point for the startup, additional tracking must be done within 20s after the first Activity calls onResume, as shown below:

Refer to the code:

```
/**
 * Users who need to customize the end point must do so within 20s after onResume.
 */
QAPM.endScene(StageConstant.QAPM_APPLAUNCH, QAPM.ModeResource);
```

Verifying Whether the Feature Is Working Properly

If the following log message appears 20 seconds after each startup or switch to background, it indicates successful reporting of startup metric data.
Retrieval tag: [plugin::114]



```
2022-05-18 15:42:55.314 23869-23918/com.example.sdkapp I/QAPM_base_JsonUploadRunnable: [plugin::114] report success
```

Note:
It requires the qapm-plugin for instrumentation and you must manually add an Application subclass. Otherwise, it will not work.
Individual event data will be reported only if the total startup time exceeds 2.5s.
Launch issue data can be viewed in Mobile App Performance Monitoring > Startup > Issue List.

Calculation

Cold Startup:

Occurs after the app starts from the device or after the system terminates the app for the first time.
Android calculation method: mainActivityOnResume_end - attachBaseContext_start.
iOS calculation method: Time of the first frame of the first page UI displayed on screen - App process creation time.

Initial Startup:

Indicates the first launch after the app installation, which is a special case of cold startup.
Android calculation method: mainActivityOnResume_end - attachBaseContext_start
iOS calculation method: Time of the first frame of the first page UI displayed on screen - App process creation time.

Warm Startup:

Under the premise that process and Activity instances still exist (for iOS, the app is in the background and alive). If the app switches to the background for three minutes and then switches back to the foreground, it is defined as a warm startup.
Android calculation method: activityOnResume_end - activityOnRestart_start.
iOS calculation method: ApplicationDidBecomeActive - ApplicationWillEnterForeground.

Startup Duration:

Total boot time / Total boot count

iOS Use Cases

Integration and Initialization

Last updated：2024-05-20 10:47:31

SDK Integration

Manual Integration

1. Download SDK.
2. Drag and drop the QAPM.framework file into the Xcode project (check the Copy items if needed option).
3. Go to TARGETS > Build Phases - Link Binary Libraries to add the dependent library:

libc++.dylib (libc++.tbd)

libz.dylib (libz.tbd)

libresolv.tbd

4. In the project's Other LinkerFlags, add the parameter `-ObjC` .
5. Import the configuration file:

For versions before 5.3.5, import the `js_sdk.js` file from the framework into the project root directory;

For versions 5.3.5 and later, import the `QAPMResourceFile.bundle` file from the framework into the project root directory.

CocoaPods Integration

In the podfile, after the following operation, execute the pod install command:

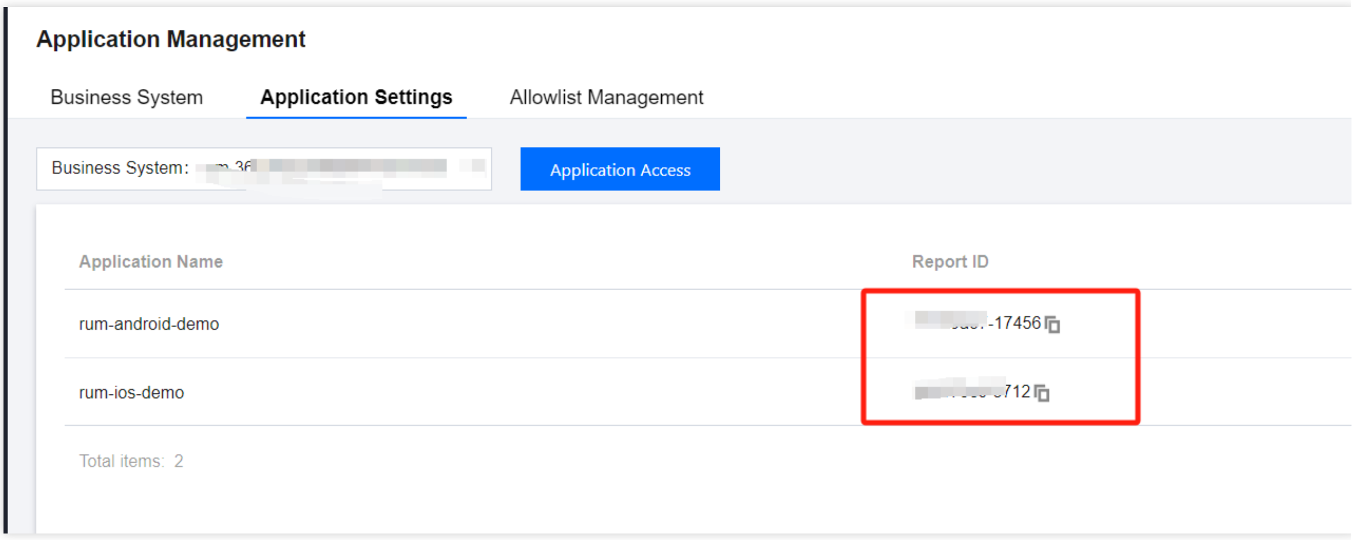
```
pod 'QAPM',:source => 'https://github.com/TencentCloud/QAPM-iOS-CocoaPods.git'
```

Note:

The minimum compatible system version for iOS SDK is iOS 8.0.

Web Environment Configurations

Log in to TCOP, on the **Mobile App Performance Monitoring** page, select **Application Management** > **Application Settings**, and enter **Application Settings** to obtain the Appkey (Report ID).


Application Management

Business System | Application Settings | Allowlist Management

Business System: [redacted]

Application Access

Application Name	Report ID
rum-android-demo	[redacted]-17456
rum-ios-demo	[redacted]712

Total items: 2

SDK Initialization

1. In the AppDelegate.m file of the project, import the header file: `#import <QAPM/QAPM.h>` . If it is a Swift project, import it in the corresponding `bridging-header.h` .
2. Initialize QAPM in the `application:didFinishLaunchingWithOptions` method of AppDelegate.m:

```
void loggerFunc(QAPMLoggerLevel level, const char* log) {

#ifdef RELEASE
    if (level <= QAPMLogLevel_Event) { ///Public release log
        NSLog(@"%@", [NSString stringWithUTF8String:log]);
}
#endif

#ifdef GRAY
    if (level <= QAPMLogLevel_Info) { ///Grayscale and public release log
        NSLog(@"%@", [NSString stringWithUTF8String:log]);
}
#endif

#ifdef DEBUG
    if (level <= QAPMLogLevel_Debug) { ///Beta, grayscale and public release log
        NSLog(@"%@", [NSString stringWithUTF8String:log]);
}
#endif
}

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSD
//In response to the Ministry of Industry and Information Technology's notice on fu
//Optional personal information includes but is not limited to device manufacturer,
//The default setting is YES, meaning collection is allowed. Example code is as fol
[QAPMConfig getInstance].collectOptionalFields = YES;

// Special Note: If the above setting is set to NO, some information will no longer

/// Setting QAPM Log Output
NSLog(@"qapm sdk version : %@", [QAPM sdkVersion]);
[QAPM registerLogCallback:loggerFunc];


//Report Address
//Domestic Site: https://app.rumt-zh.com//International Site: https://app.rumt-sg.c
 [QAPMConfig getInstance].host = @"https://app.rumt-sg.com";

[QAPMConfig getInstance].customerAppVersion = @"Set app custom version number";
//Based on the actual situation, set userId and deviceId.
//Unique device identifier deviceId, such as IDFV in conjunction with Keychain.
[QAPMConfig getInstance].userId = @"Set userId";
    [QAPMConfig getInstance].deviceID = @"Custom deviceId";
    /// QAPM Startup
    [QAPM startWithAppKey:@"Unique product appKey"];
    return YES;
}
```

Note:

During the development process, you can add the set user ID/device ID to the allowlist to ensure that the feature reporting of the specified device is not affected by sampling. It can be done in Mobile App Performance Monitoring > Application Management > **Allowlist Management** by clicking **Allowlist Configuration** > **Add**.

Monitoring Feature Enabled

Recommended Use:

In the API of `QAPMModelStableConfig.h` file: `- (void)setupModelAll; //This API can enable all monitoring features.`
In the API of `QAPMModelStableConfig.h` file: `- (void)setupModelStable;//This interface can enable all monitoring features except network monitoring and user behavior monitoring. According to the requirements of the Ministry of Industry and Information Technology Document No. 26, network monitoring and user behavior monitoring are classified as extended business features. You can choose whether to activate them based on your needs.`

Custom Use:

To custom enabling QAPM features (including stalling, crash, native network, native user behavior, startup, and WebView), you can set a combination of enumerated values in the API of `enableMonitorTypeOptions` in the `QAPMConfig.h` file, adding multiple parameter enumeration values to activate a series of features. The code is as follows:

```
// Setting the Monitoring Enabled by QAPM:
[QAPMConfig getInstance].enableMonitorTypeOptions =
    QAPMMonitorTypeBlue |
    QAPMMonitorTypeCrash |
    QAPMMonitorTypeHTTPMonitor |
    QAPMMonitorTypeIUPMonitor|
    QAPMMonitorTypeLaunch |
    QAPMMonitorTypeJSError |
    QAPMMonitorTypeWebViewNetWork |
    QAPMMonitorTypeWebViewIUPMonitor |
    QAPMMonitorTypeWebMonitor;
```

API Description:

API: In the `QAPMConfig.h` class `@property (nonatomic, assign) QAPMMonitorType enableMonitorTypeOptions;`

Parameter Description: `QAPMMonitorType` is a feature type, with optional values as shown in the following code.

Optional Item Description

:

```
///Stalling Detection Feature
```

```
QAPMMonitorTypeBlue
///Crash Monitoring Feature
QAPMMonitorTypeCrash
///Native Network Monitoring
QAPMMonitorTypeHTTPMonitor
///Native User Behavior Monitoring
QAPMMonitorTypeIUPMonitor
///Case Monitoring Feature Startup
QAPMMonitorTypeLaunch
///WebView's JS Exception Monitoring
QAPMMonitorTypeJSError
///WebView's Network
QAPMMonitorTypeWebViewNetWork
///WebView User Behavior Monitoring
QAPMMonitorTypeWebViewIUPMonitor
///WebView Page Performance Monitoring
QAPMMonitorTypeWebMonitor
```

Note:

1. Use the OR operation to custom enabling the required monitoring features, for example, for stalling: `QAPMMonitorTypeBlue` .
2. **In response to the Ministry of Industry and Information Technology's (MIIT)** notice on further enhancing the service capabilities of mobile Internet applications**, we have divided network monitoring and user behavior monitoring into**

, based on the MIIT's definition of basic features for performance monitoring SDKs. To avoid collecting personal information such as network log information and user operation records, you can selectively enable these two features. At the operational level, you can prevent the entry of **`QAPMMonitorTypeHTTPMonitor`** and **`QAPMMonitorTypeWebViewNetWork`** **parameters in the custom performance module configuration to disable the network monitoring feature. Similarly, you can prevent the entry of** **`QAPMMonitorTypeIUPMonitor`** **and** **`QAPMMonitorTypeWebViewIUPMonitor`** **parameters to disable the user behavior monitoring feature, or simply use the ModelStable feature activation mode to automatically enable other recommended features while all extended business features are disabled.**

SDK Feature Introduction

Stalling and Smoothness Monitoring

Stalling Detection Feature

QAPMMoniterType: QAPMMonitorTypeBlue Stalling detection feature collects and reports stack informaiton if the stalling duration exceeds the 200 ms threshold.

Smoothness Monitoring

To start monitoring smoothness statistics in swipe scenarios, add the following codes to the relevant pages. Logs will be reported upon the next app startup.

```
#pragma mark - TableView Delegate
- (void)scrollViewWillBeginDragging:(UIScrollView *)scrollView {
    [QAPMBlueProfile beginTrackingWithStage:NSStringFromClass([self class])];
}

- (void)scrollViewDidEndDragging:(UIScrollView *)scrollView willDecelerate:(BOOL)de
    if(!decelerate){
        [QAPMBlueProfile stopTrackingWithStage:NSStringFromClass([self class])];
    }
    }

    - (void)scrollViewDidEndDecelerating:(UIScrollView *)scrollView {
        [QAPMBlueProfile stopTrackingWithStage:NSStringFromClass([self class])];
    }
```

Crash Monitoring

QAPMMonitorTypeCrash Crash logs will be reported upon the next SDK startup.

Note:

1. If the business side is using a third-party SDK for collecting normal crash data, please uninstall the third-party monitoring software to avoid reporting inaccurate stack information.
2. After a FOOM or deadlock freeze exit, the related stack information recorded from the last session will be reported upon the next startup. FOOM are fully reported in debug or non-App Store scenarios. In the App Store environment, data is sampled at 2%.

Startup Duration Monitoring

Description

By using the startup duration monitoring feature, you can measure the duration from the creation of the app process to the first frame of the UI being rendered.

When the startup time exceeds the threshold (4,000 ms by default), detailed case information is reported. This includes the startup duration, automatic tagging interval, custom tagging intervals, and the startup process stack information.

Related APIs

```
@interface QAPMLaunchProfile : NSObject
/**
Set a custom tagging interval for begin. This interval must be within the startup t
 @param scene Scene Name
 */
- (void)setBeginTimestampForScene:(NSString *)scene;
/**
Set a custom tagging interval for end. This interval must be within the startup tim
 @param scene Scene Name
 */
- (void)setEndTimestampForScene:(NSString *)scene;
```

```
@end
```

Sample Code

In the corresponding class of the project, import the header file `#import <QAPM/QAPMLaunchProfile.h>` .

Starting monitoring component in the main function:

```
int main(int argc, char * argv[]) {

    @autoreleasepool {

        [QAPMLaunchProfile didEnterMain];
        return UIApplicationMain(argc, argv, nil, NSStringFromClass([AppDelegate cl
    }
}
```

Web Monitoring

Note:

1. Currently, WKWebView supports iOS 11 and later.
2. This feature can monitor the loading time of web network resources and jserror.

Feature Configurations

Web Configurations

Note:

If you use manual integration, you need to import the js_sdk inside the framework into the project using the Add Files to method. If integrated with CocoaPods, this step is not needed. If you use the TMFWebOffline offline package feature, you need to import #import <TMFQWebView/QBWKWebView.h> into the header file of the related pages in the project and adhere to the TMFWebOfflineWebViewControllerProtocol proxy. WKWebView inherits TMFWkWebView as set below:

```
#import <TMFQWebView/QBWKWebView.h>

@interface WKWebviewViewController ()<WKUIDelegate, WKNavigationDelegate, WKScriptM
 {
    TMFWkWebView *wkWebView;
}
```

iOS SDK Configurations

Add `#import <QAPM/QAPMLaunchProfile.h>` in the class file to import the SDK header file.

Add the following code in the WKWebView proxy method of webView:didFinishNavigation to provide a Web API for obtaining native-related information.

```
- (void)webView:(WKWebView *)webView didFinishNavigation:(WKNavigation *)navigation
 {
[webView evaluateJavaScript:[QAPMWebViewProfile qapmBaseInfo:@" "] completionHandle
 [webView evaluateJavaScript:[QAPMWebViewProfile qapmJsStart] completionHandler:nil
}
```

User Privacy Agreement

Privacy compliance policy: Due to privacy compliance requirements, ensure not to call any QAPM APIs before users' consent for privacy compliance. In addition, QAPM still requires a device-level unique identifier to determine the uniqueness of the device, used for user-level metric calculation.

```
// QAPM can only be initialized normally after the user has granted authorization
if (isAgree) {        }
//The first tagging for the startup duration function
// Requires passing in a unique device identifier, such as IDFV in conjunction with
```

```
[QAPMConfig getInstance].deviceID = @"Custom deviceId";
//User ID or third-party log-in account. This API can be used multiple times in the
[QAPMConfig getInstance].userId = @"Set userId";
```

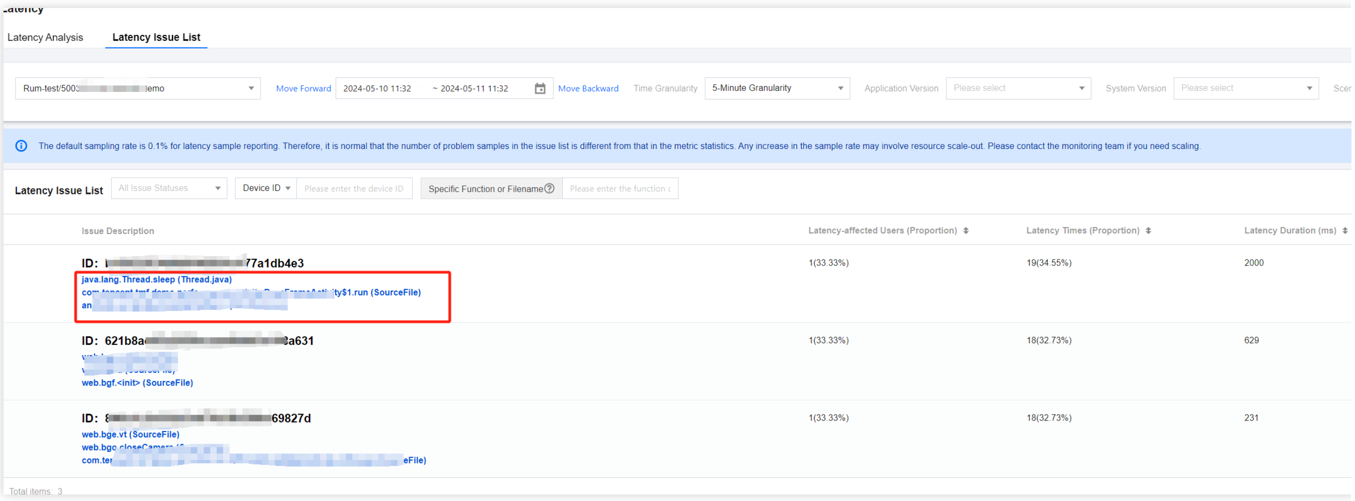
Note:

The background of deviceID collection method change: The current regulations require the SDK not to collect information such as UDID directly or indirectly. We can only let users input identifiers to distinguish different devices. This effectively reduces the distortions in crash rate metric data.

Symbol Table Configurations

Use the reported individual case data to upload the symbol table. Identify the corresponding symbol table via the build ID on the individual case page. You can use the official atos command to translate a certain line of stack on the individual case page to ensure the symbol table and translation are normal.

1. There is an uploading entrance on the individual case page. For example, select **Latency** > **Latency Analysis** > **Latency Issue List**, to enter the details page.



Rum-test/5003[redacted]emo

2024-05-10 11:32 ~ 2024-05-11 11:32

5-Minute Granularity

Please select

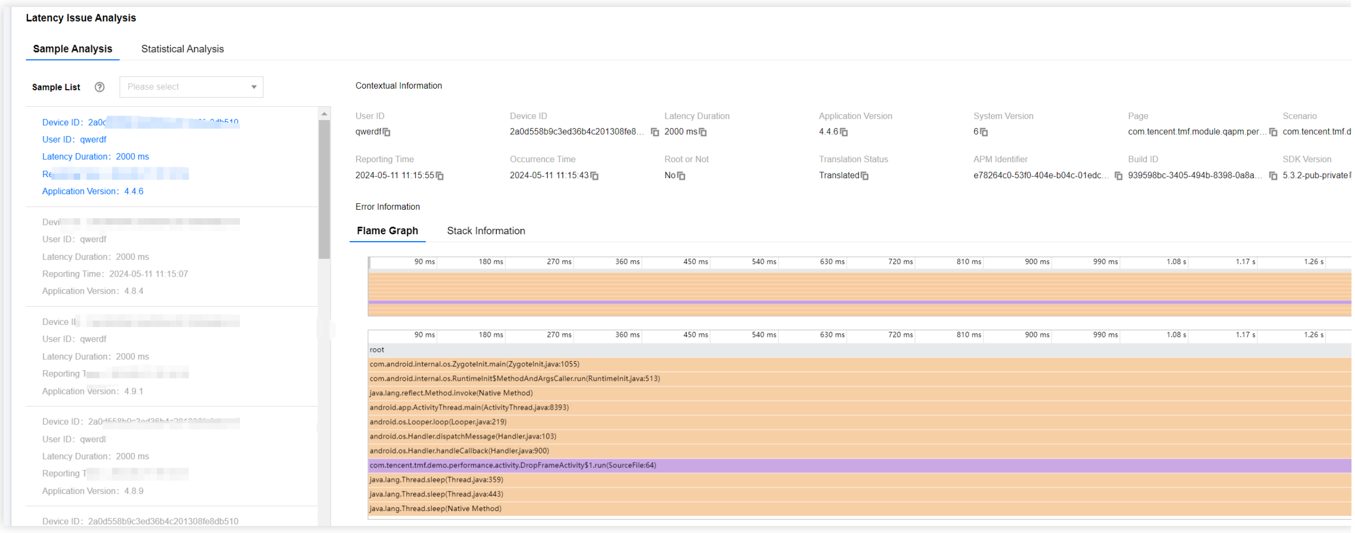
The default sampling rate is 0.1% for latency sample reporting. Therefore, it is normal that the number of problem samples in the issue list is different from that in the metric statistics. Any increase in the sample rate may involve resource scale-out. Please contact the monitoring team if you need scaling.

Latency Issue List | All Issue Statuses | Device ID | Please enter the device ID | Specific Function or Filename | Please enter the function c

Issue Description	Latency-affected Users (Proportion)	Latency Times (Proportion)	Latency Duration (ms)
ID: [redacted]77a1db4e3 java.lang.Thread.sleep (Thread.java) com[redacted]ty$1.run (SourceFile) an[redacted]	1(33.33%)	19(34.55%)	2000
ID: 621b8a[redacted]a631 [redacted] web.bgf.<init> (SourceFile)	1(33.33%)	18(32.73%)	629
ID: [redacted]69827d web.bge.vt (SourceFile) web.bgc[redacted] com.te[redacted]eFile)	1(33.33%)	18(32.73%)	231

Total items: 3

2. On the details page, click **Upload Mapping/.so File..**



Device ID: 2a0c[redacted]
User ID: qwerdf
Latency Duration: 2000 ms
Re[redacted]
Application Version: 4.4.6

Devi[redacted]
User ID: qwerdf
Latency Duration: 2000 ms
Reporting Time: 2024-05-11 11:15:07
Application Version: 4.8.4

Device I[redacted]
User ID: qwerdf
Latency Duration: 2000 ms
Reporting T[redacted]
Application Version: 4.9.1

Device ID: 2a0[redacted]
User ID: qwerd[redacted]
Latency Duration: 2000 ms
Reporting T[redacted]
Application Version: 4.8.9

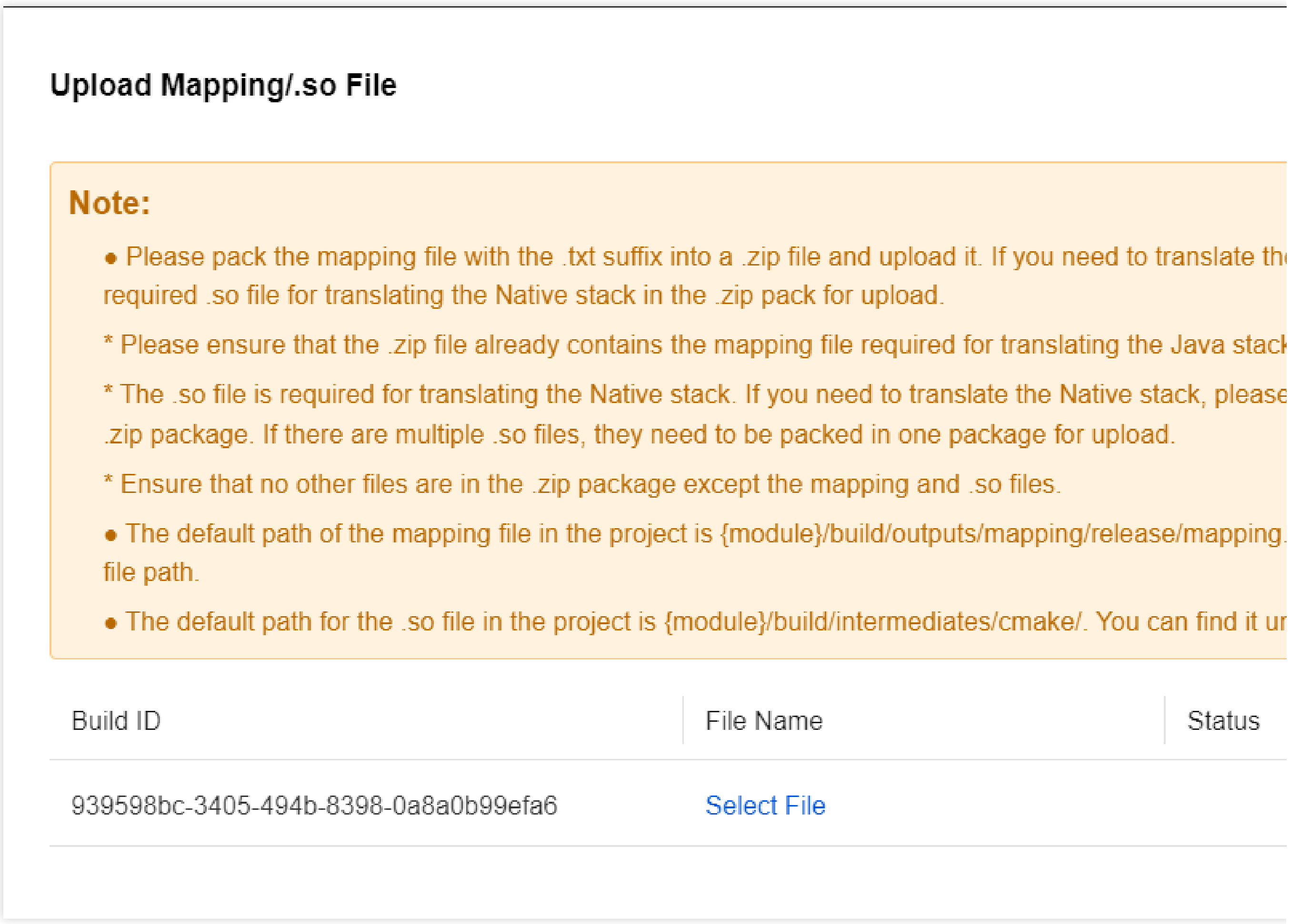
Device ID: 2a0d558b9c3ed36b4c201308fe8db510

Contextual Information

User ID	Device ID	Latency Duration	Application Version	System Version	Page	Scenario
qwerdf	2a0d558b9c3ed36b4c201308fe8...	2000 ms	4.4.6	6	com.tencent.tmf.module.qapm.per...	com.tencent.tmf.d
Reporting Time	**Occurrence Time**	**Root or Not**	**Translation Status**	**APM Identifier**	**Build ID**	**SDK Version**
2024-05-11 11:15:55	2024-05-11 11:15:43	No	Translated	e78264c0-53f0-404e-b04c-01edc...	939598bc-3405-494b-8398-0a8a...	5.3.2-pub-private

```
root
com.android.internal.os.ZygoteInit.main(ZygoteInit.java:1055)
com.android.internal.os.RuntimeInit$MethodAndArgsCaller.run(RuntimeInit.java:513)
java.lang.reflect.Method.invoke(Native Method)
android.app.ActivityThread.main(ActivityThread.java:8393)
android.os.Looper.loop(Looper.java:219)
android.os.Handler.dispatchMessage(Handler.java:103)
android.os.Handler.handleCallback(Handler.java:900)
com.tencent.tmf.demo.performance.activity.DropFrameActivity$1.run(SourceFile:64)
java.lang.Thread.sleep(Thread.java:359)
java.lang.Thread.sleep(Thread.java:443)
java.lang.Thread.sleep(Native Method)
```

3. Go to the symbol table upload page, click **Select File**, then select the dSYM file corresponding to that build.



Upload Mapping/.so File

Note:

- Please pack the mapping file with the .txt suffix into a .zip file and upload it. If you need to translate th required .so file for translating the Native stack in the .zip pack for upload.

* Please ensure that the .zip file already contains the mapping file required for translating the Java stack

* The .so file is required for translating the Native stack. If you need to translate the Native stack, please .zip package. If there are multiple .so files, they need to be packed in one package for upload.

* Ensure that no other files are in the .zip package except the mapping and .so files.

- The default path of the mapping file in the project is {module}/build/outputs/mapping/release/mapping. file path.
- The default path for the .so file in the project is {module}/build/intermediates/cmake/. You can find it ur

Build ID	File Name	Status
939598bc-3405-494b-8398-0a8a0b99efa6	Select File	

Viewing QAPM Work Log

Setting Work Log Viewing

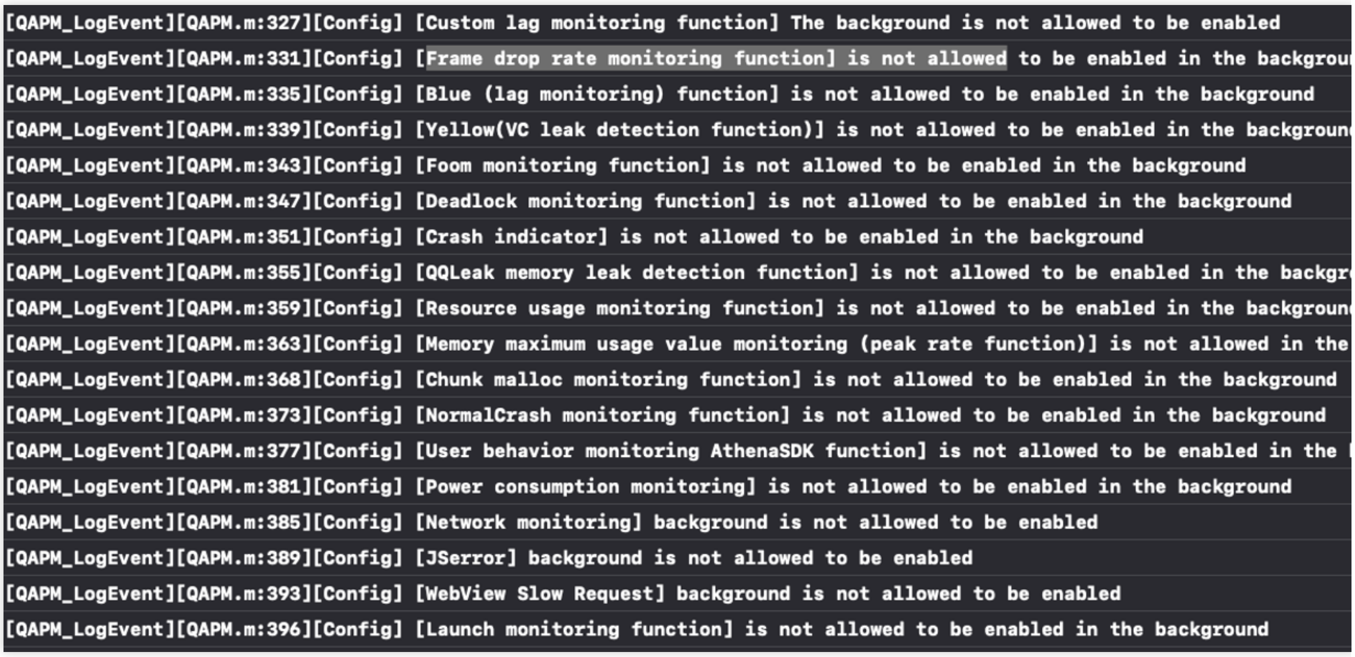
Before you start the QAPM SDK with `[QAPM startWithAppKey:]` , set the log output function to control the log output according to different release versions:

```
void loggerFunc(QAPMLoggerLevel level, const char* log) {

#ifdef RELEASE
    if (level <= QAPMLogLevel_Event) { ///Public release log
        NSLog(@"%@", [NSString stringWithUTF8String:log]);
}
#endif

#ifdef GRAY
    if (level <= QAPMLogLevel_Info) { ///Grayscale and public release log
        NSLog(@"%@", [NSString stringWithUTF8String:log]);
}
#endif

#ifdef DEBUG
    if (level <= QAPMLogLevel_Debug) { ///Beta, grayscale and public release log
        NSLog(@"%@", [NSString stringWithUTF8String:log]);
}
#endif
}

- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSD

    /// Setting QAPM Log Output
    [QAPM registerLogCallback:loggerFunc];

    /// ...
    /// Setting the QAPM SDK Startup
}
```

Submitting Log Analysis

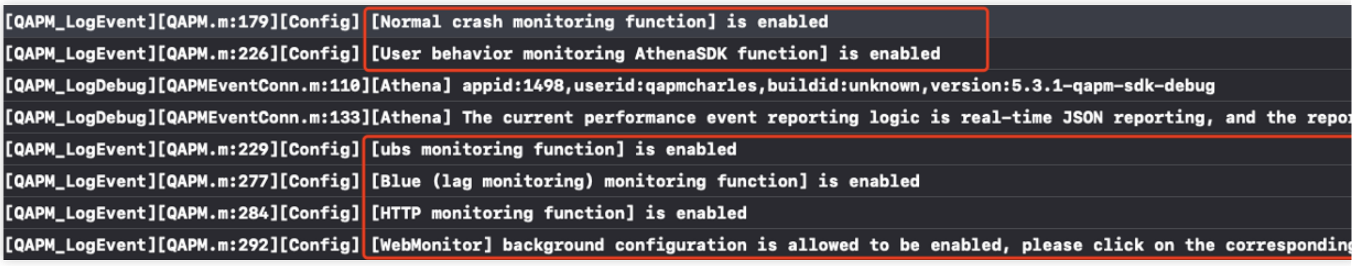
After the SDK is connected, the monitoring features are usually confirmed to be enabled by analyzing logs.

When the monitoring feature is not enabled, the logs are as follows:



```
[QAPM_LogEvent][QAPM.m:327][Config] [Custom lag monitoring function] The background is not allowed to be enabled
[QAPM_LogEvent][QAPM.m:331][Config] [Frame drop rate monitoring function] is not allowed to be enabled in the backgrou
[QAPM_LogEvent][QAPM.m:335][Config] [Blue (lag monitoring) function] is not allowed to be enabled in the background
[QAPM_LogEvent][QAPM.m:339][Config] [Yellow(VC leak detection function)] is not allowed to be enabled in the backgroun
[QAPM_LogEvent][QAPM.m:343][Config] [Foom monitoring function] is not allowed to be enabled in the background
[QAPM_LogEvent][QAPM.m:347][Config] [Deadlock monitoring function] is not allowed to be enabled in the background
[QAPM_LogEvent][QAPM.m:351][Config] [Crash indicator] is not allowed to be enabled in the background
[QAPM_LogEvent][QAPM.m:355][Config] [QQLeak memory leak detection function] is not allowed to be enabled in the backgr
[QAPM_LogEvent][QAPM.m:359][Config] [Resource usage monitoring function] is not allowed to be enabled in the backgroun
[QAPM_LogEvent][QAPM.m:363][Config] [Memory maximum usage value monitoring (peak rate function)] is not allowed in the
[QAPM_LogEvent][QAPM.m:368][Config] [Chunk malloc monitoring function] is not allowed to be enabled in the background
[QAPM_LogEvent][QAPM.m:373][Config] [NormalCrash monitoring function] is not allowed to be enabled in the background
[QAPM_LogEvent][QAPM.m:377][Config] [User behavior monitoring AthenaSDK function] is not allowed to be enabled in the
[QAPM_LogEvent][QAPM.m:381][Config] [Power consumption monitoring] is not allowed to be enabled in the background
[QAPM_LogEvent][QAPM.m:385][Config] [Network monitoring] background is not allowed to be enabled
[QAPM_LogEvent][QAPM.m:389][Config] [JSerror] background is not allowed to be enabled
[QAPM_LogEvent][QAPM.m:393][Config] [WebView Slow Request] background is not allowed to be enabled
[QAPM_LogEvent][QAPM.m:396][Config] [Launch monitoring function] is not allowed to be enabled in the background
```

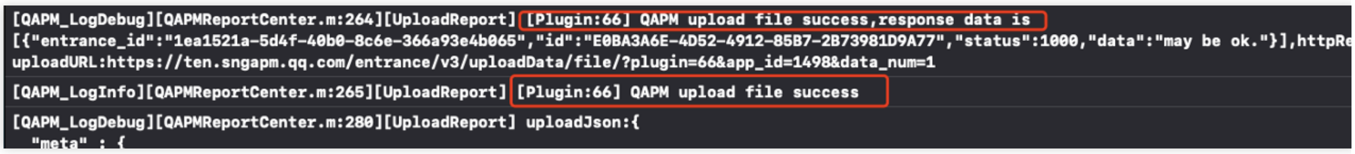
When the monitoring feature is enabled, the logs are as follows:



```
[QAPM_LogEvent][QAPM.m:179][Config] [Normal crash monitoring function] is enabled
[QAPM_LogEvent][QAPM.m:226][Config] [User behavior monitoring AthenaSDK function] is enabled
[QAPM_LogDebug][QAPMEventConn.m:110][Athena] appid:1498,userid:qapmcharles,buildid:unknown,version:5.3.1-qapm-sdk-debug
[QAPM_LogDebug][QAPMEventConn.m:133][Athena] The current performance event reporting logic is real-time JSON reporting, and the repo
[QAPM_LogEvent][QAPM.m:229][Config] [ubs monitoring function] is enabled
[QAPM_LogEvent][QAPM.m:277][Config] [Blue (lag monitoring) monitoring function] is enabled
[QAPM_LogEvent][QAPM.m:284][Config] [HTTP monitoring function] is enabled
[QAPM_LogEvent][QAPM.m:292][Config] [WebMonitor] background configuration is allowed to be enabled, please click on the corresponding
```

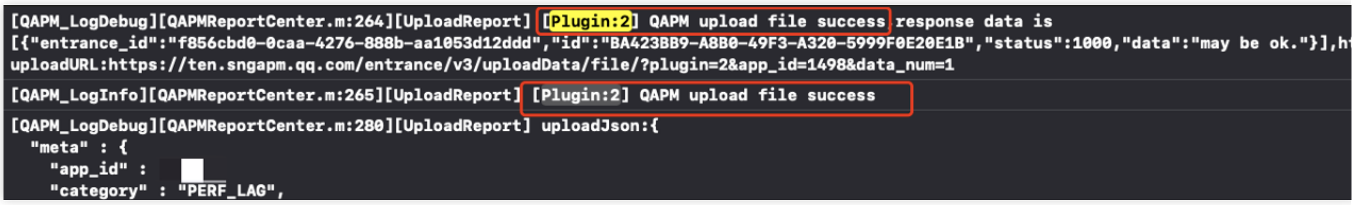
By checking the initialization logs, you can confirm whether the initialization is successful. All monitoring features are enabled, and then each feature's successful reporting is verified.

Startup Duration Report



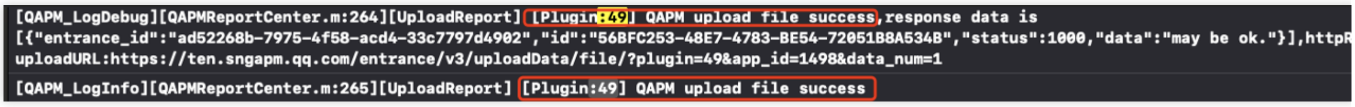
```
[QAPM_LogDebug][QAPMReportCenter.m:264][UploadReport] [Plugin:66] QAPM upload file success,response data is
[{"entrance_id":"1ea1521a-5d4f-40b0-8c6e-366a93e4b065","id":"E0BA3A6E-4D52-4912-85B7-2B73981D9A77","status":1000,"data":"may be ok."}],httpRe
uploadURL:https://ten.sngapm.qq.com/entrance/v3/uploadData/file/?plugin=66&app_id=1498&data_num=1
[QAPM_LogInfo][QAPMReportCenter.m:265][UploadReport] [Plugin:66] QAPM upload file success
[QAPM_LogDebug][QAPMReportCenter.m:280][UploadReport] uploadJson:{
  "meta" : {
```

Stalling Case Report



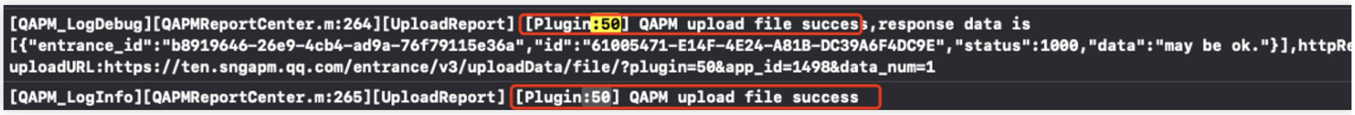
```
[QAPM_LogDebug][QAPMReportCenter.m:264][UploadReport] [Plugin:2] QAPM upload file success,response data is
[{"entrance_id":"f856cbd0-0caa-4276-888b-aa1053d12ddd","id":"BA423BB9-A8B0-49F3-A320-5999F0E20E1B","status":1000,"data":"may be ok."}],h
uploadURL:https://ten.sngapm.qq.com/entrance/v3/uploadData/file/?plugin=2&app_id=1498&data_num=1
[QAPM_LogInfo][QAPMReportCenter.m:265][UploadReport] [Plugin:2] QAPM upload file success
[QAPM_LogDebug][QAPMReportCenter.m:280][UploadReport] uploadJson:{
  "meta" : {
    "app_id" : [redacted]
    "category" : "PERF_LAG",
```

FOOM Case Report



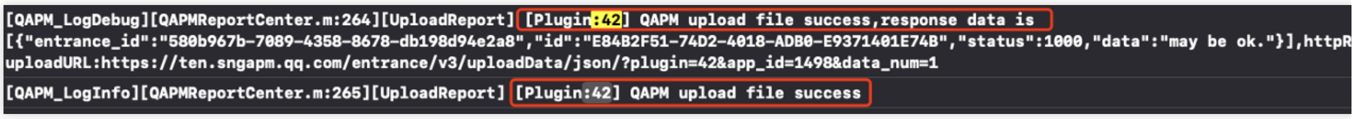
```
[QAPM_LogDebug][QAPMReportCenter.m:264][UploadReport] [Plugin:49] QAPM upload file success,response data is
[{"entrance_id":"ad52268b-7975-4f58-acd4-33c7797d4902","id":"56BFC253-48E7-4783-BE54-72051B8A534B","status":1000,"data":"may be ok."}],httpR
uploadURL:https://ten.sngapm.qq.com/entrance/v3/uploadData/file/?plugin=49&app_id=1498&data_num=1
[QAPM_LogInfo][QAPMReportCenter.m:265][UploadReport] [Plugin:49] QAPM upload file success
```

Deadlock Case Report



```
[QAPM_LogDebug][QAPMReportCenter.m:264][UploadReport] [Plugin:50] QAPM upload file success,response data is
[{"entrance_id":"b8919646-26e9-4cb4-ad9a-76f79115e36a","id":"61005471-E14F-4E24-A81B-DC39A6F4DC9E","status":1000,"data":"may be ok."}],httpRe
uploadURL:https://ten.sngapm.qq.com/entrance/v3/uploadData/file/?plugin=50&app_id=1498&data_num=1
[QAPM_LogInfo][QAPMReportCenter.m:265][UploadReport] [Plugin:50] QAPM upload file success
```

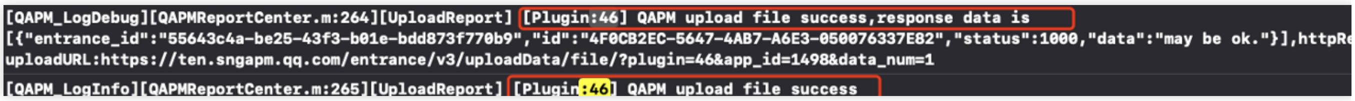
HTTP Monitoring Report



```
[QAPM_LogDebug][QAPMReportCenter.m:264][UploadReport] [Plugin:42] QAPM upload file success,response data is
[{"entrance_id":"580b967b-7089-4358-8678-db198d94e2a8","id":"E84B2F51-74D2-4018-ADB0-E9371401E74B","status":1000,"data":"may be ok."}],httpR
uploadURL:https://ten.sngapm.qq.com/entrance/v3/uploadData/json/?plugin=42&app_id=1498&data_num=1
[QAPM_LogInfo][QAPMReportCenter.m:265][UploadReport] [Plugin:42] QAPM upload file success
```

Report of a Normal Crash

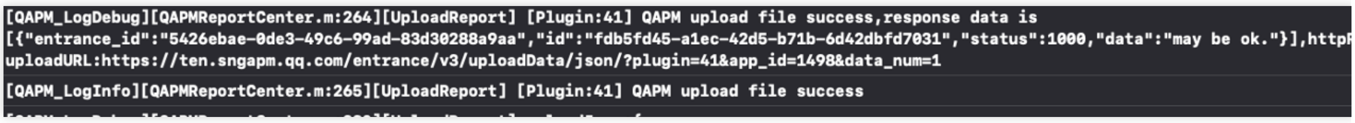
When the report of a normal crash is being triggered, do not connect the data cable to Xcode. After the normal crash is triggered, the report information can be seen upon the restart of the App. This reporting log can be viewed using the console that comes with Mac, and the log is as follows:



```
[QAPM_LogDebug][QAPMReportCenter.m:264][UploadReport] [Plugin:46] QAPM upload file success,response data is
[{"entrance_id":"55643c4a-be25-43f3-b01e-bdd873f770b9","id":"4F0CB2EC-5647-4AB7-A6E3-050076337E82","status":1000,"data":"may be ok."}],httpRe
uploadURL:https://ten.sngapm.qq.com/entrance/v3/uploadData/file/?plugin=46&app_id=1498&data_num=1
[QAPM_LogInfo][QAPMReportCenter.m:265][UploadReport] [Plugin:46] QAPM upload file success
```

Webview and JSerror Report

Webview and jserror report can be viewed in Xcode, based on plugin:43 and plugin:41.



```
[QAPM_LogDebug][QAPMReportCenter.m:264][UploadReport] [Plugin:41] QAPM upload file success,response data is
[{"entrance_id":"5426ebae-0de3-49c6-99ad-83d30288a9aa","id":"fdb5fd45-a1ec-42d5-b71b-6d42dbfd7031","status":1000,"data":"may be ok."}],httpR
uploadURL:https://ten.sngapm.qq.com/entrance/v3/uploadData/json/?plugin=41&app_id=1498&data_num=1
[QAPM_LogInfo][QAPMReportCenter.m:265][UploadReport] [Plugin:41] QAPM upload file success
```

Note:

For more advanced feature configurations, see GitHub - TencentCloud/qapm-sdk-ios. It is mainly used for updating the QAPM performance monitoring component, as well as documentation in the Shell script and related documentation folder.

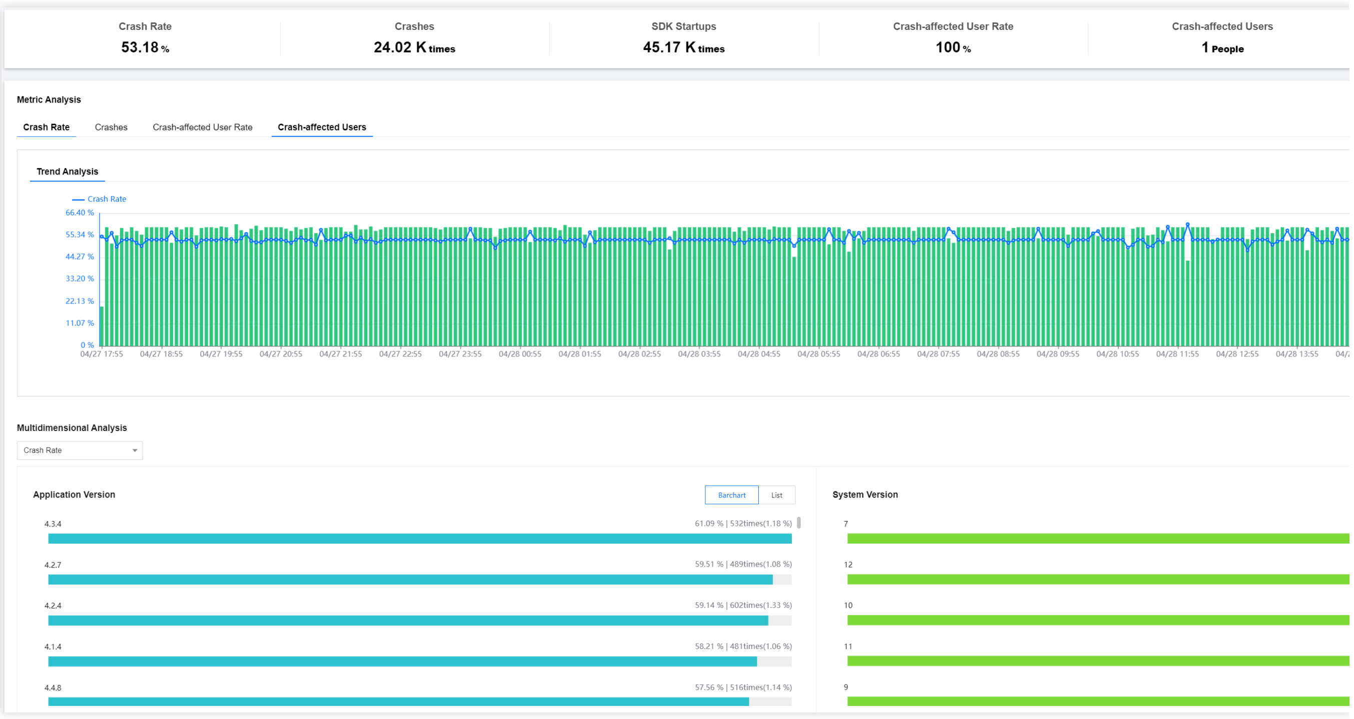
Operation Guide
Crash

Last updated：2024-05-13 18:03:25

Terminal Performance Monitoring aggregates key characteristics from individual crash incidents, facilitating root cause analysis of App crashes.

Feature Entry

1. Log in to Tencent Cloud Observability Platform.
2. In the left navigation bar, select **Mobile App Performance Monitoring > Crash**. Select the business system, app, and time range to analyze crashes.



Crash Rate	Crashes	SDK Startups	Crash-affected User Rate	Crash-affected Users
53.18 %	24.02 K times	45.17 K times	100 %	1 People

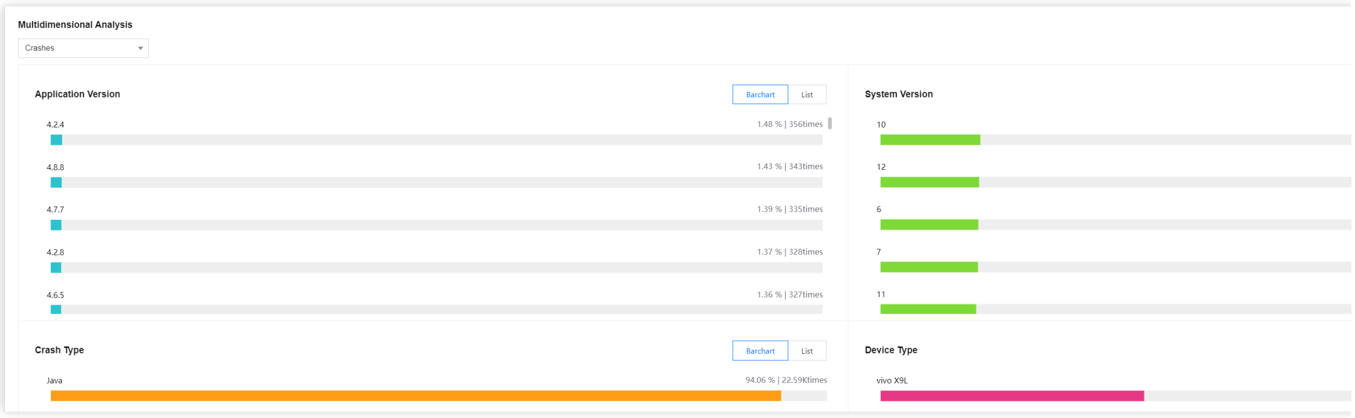
Metric Analysis

Application Version

4.3.4	61.09 % | 532times(1.18 %)
4.2.7	59.51 % | 489times(1.08 %)
4.2.4	59.14 % | 602times(1.33 %)
4.1.4	58.21 % | 481times(1.06 %)
4.4.8	57.56 % | 516times(1.14 %)

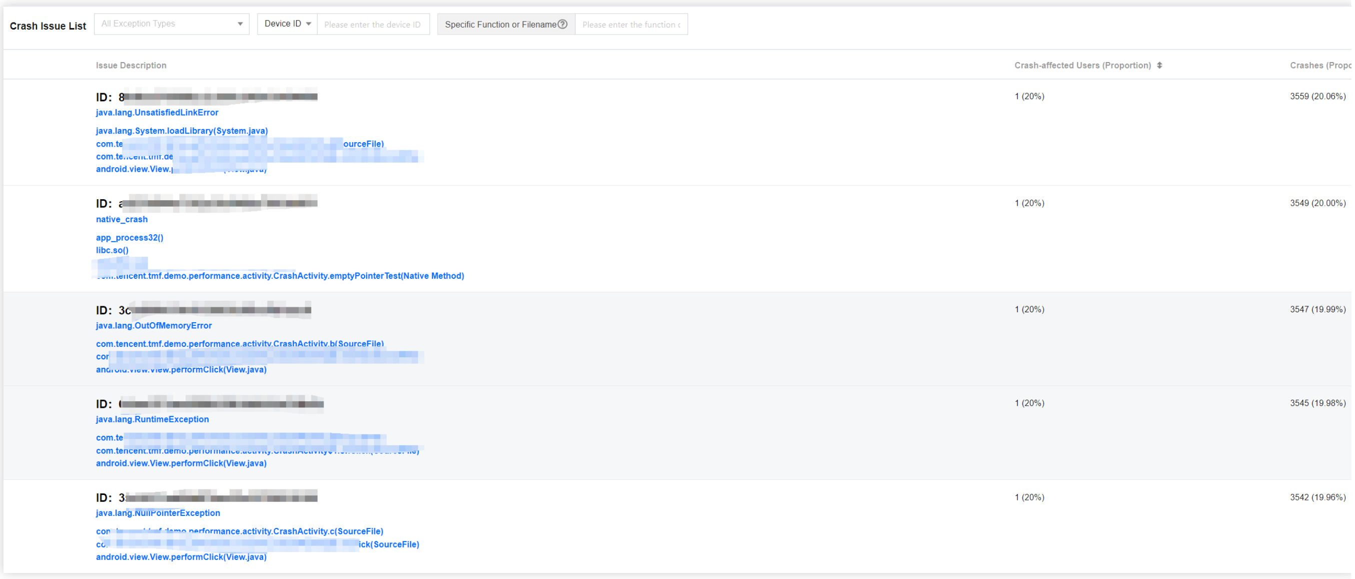
Multidimensional Analysis

The Multidimensional Analysis page shows the analysis of key metrics from multiple dimensions such as app version, system version, crash type, device type, and app status. It facilitates targeted root cause analysis of specific crash events.



Crash Issue List

The crash issue list shows crashes of all devices. You can quickly filter crashes by error type, device ID, function, or filename. You can also click **Issue Description** to view the details of crashes and pinpoint the root causes of crashes.


Crash Issue List
All Exception Types
Device ID
Please enter the device ID
Specific Function or Filename
Please enter the function

Issue Description	Crash-affected Users (Proportion)	Crashes (Propo
ID: 8[redacted] java.lang.UnsatisfiedLinkError java.lang.System.loadLibrary(System.java) com.te[redacted]ourceFile) com.te[redacted] android.view.View.[redacted]	1 (20%)	3559 (20.06%)
ID: a[redacted] native_crash app_process32() libc.so() [redacted] [redacted]encent.tmf.demo.performance.activity.CrashActivity.emptyPointerTest(Native Method)	1 (20%)	3549 (20.00%)
ID: 3c[redacted] java.lang.OutOfMemoryError com.tencent.tmf.demo.performance.activity.CrashActivity.b(SourceFile) cor[redacted] and[redacted]iew.View.performClick(View.java)	1 (20%)	3547 (19.99%)
ID: [redacted] java.lang.RuntimeException com.te[redacted] com.tencent.tmf.demo.performance.activity.CrashActivity[redacted] android.view.View.performClick(View.java)	1 (20%)	3545 (19.98%)
ID: 3[redacted] java.lang.NullPointerException con[redacted]demo.performance.activity.CrashActivity.c(SourceFile) co[redacted]ick(SourceFile) android.view.View.performClick(View.java)	1 (20%)	3542 (19.96%)

Metrics Description

Related Metrics are as follows:

Metric Name	Metrics Description

Crash Rate	Number of crashes/Number of app launches within a specified time range
Crash-affected User Rate	Number of users affected by crashes/Number of users launching the app within a specified time range
Crashes	Number of crashes within a specified time range
Crash-affected Users	Number of users affected by crashes within a specified time range
Crash Type	Classified into Java crashes and Native crashes based on the location of the crash occurrence
SDK Startups	Number of app launches

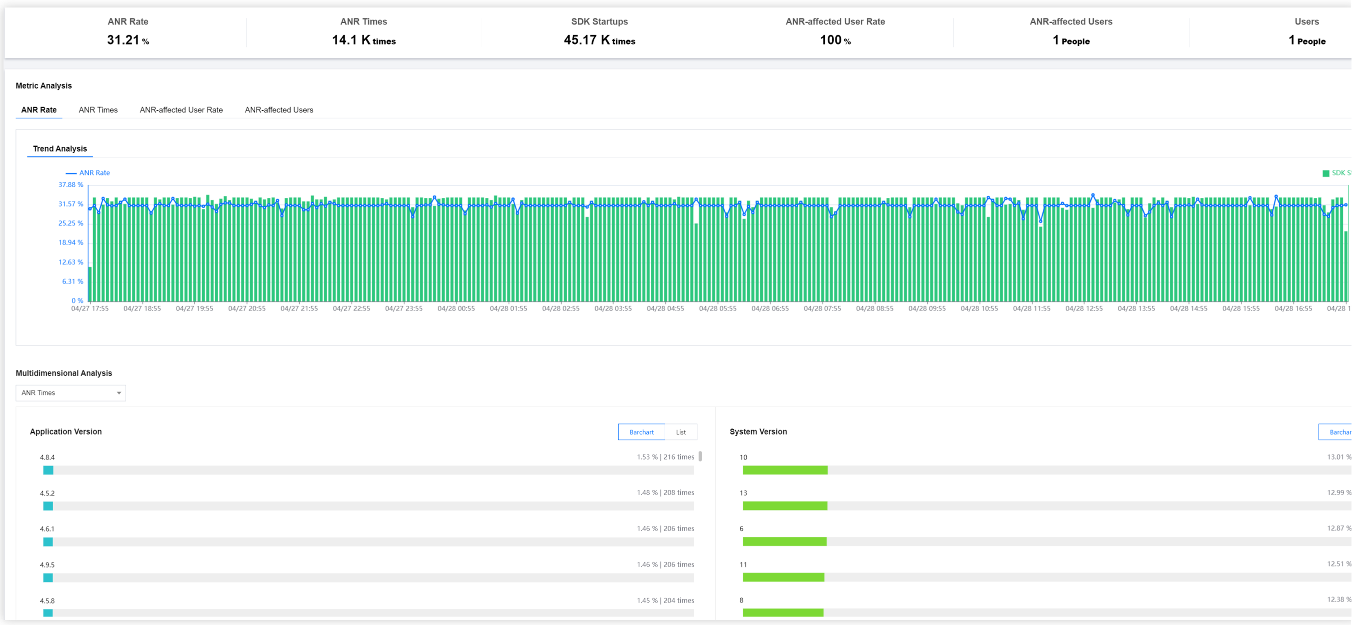
ANR

Last updated：2024-05-13 18:03:25

Mobile Performance Monitoring (MPM) aggregates key characteristics of individual Application Not Response (ANR) cases, facilitating root cause analysis of ANR issues for your app.

Feature Entry

1. Log in to Tencent Cloud Observability Platform.
2. In the left sidebar menu, select **Mobile App Performance Monitoring** > **Anr**.
3. Select the business system, app, and time range to analyze ANR issues.



ANR Rate	ANR Times	SDK Startups	ANR-affected User Rate	ANR-affected Users	Users
31.21 %	14.1 K times	45.17 K times	100 %	1 People	1 People

Metric Analysis

ANR Rate | ANR Times | ANR-affected User Rate | ANR-affected Users

Trend Analysis

Multidimensional Analysis

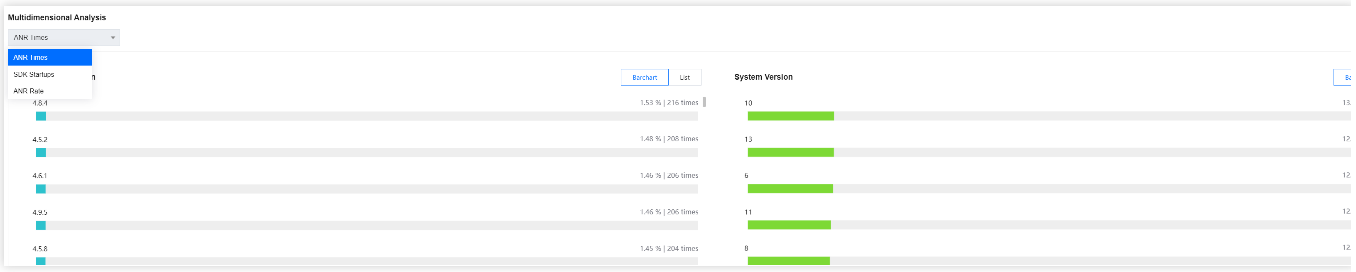
ANR Times

Application Version	
4.8.4	1.53 % | 216 times
4.5.2	1.48 % | 208 times
4.6.1	1.46 % | 206 times
4.9.5	1.46 % | 206 times
4.5.8	1.45 % | 204 times

System Version	
10	13.01 %
13	12.99 %
6	12.87 %
11	12.51 %
8	12.38 %

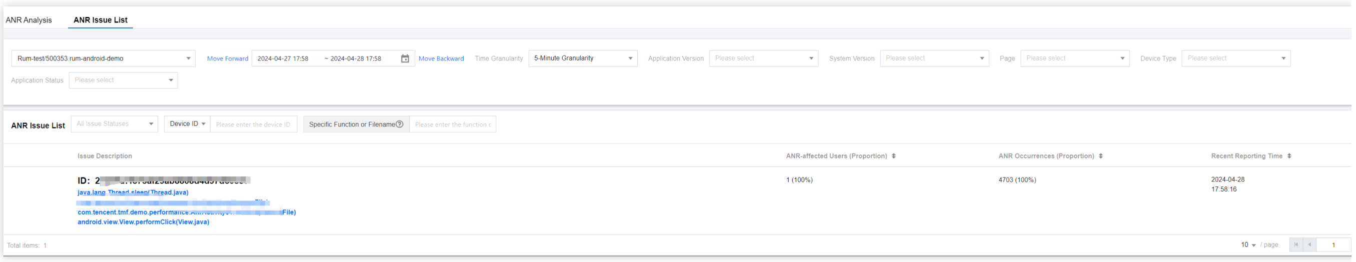
Multidimensional Analysis

The Multidimensional analysis page shows the analysis of key metrics from multiple dimensions such as ANR count, number of launches, and ANR rate. It facilitates targeted root cause analysis of specific ANR issues.



ANR Issue List

The ANR issue list allows you to view, search, sort, and manage clustering issues. You can enter the issue details page by clicking the View Details button.


ANR Analysis | **ANR Issue List**

Rum-test/500353.rum-android-demo | Move Forward | 2024-04-27 17:58 ~ 2024-04-28 17:58 | Move Backward | Time Granularity: 5-Minute Granularity | Application Version: Please select | System Version: Please select | Page: Please select | Device Type: Please select | Application Status: Please select

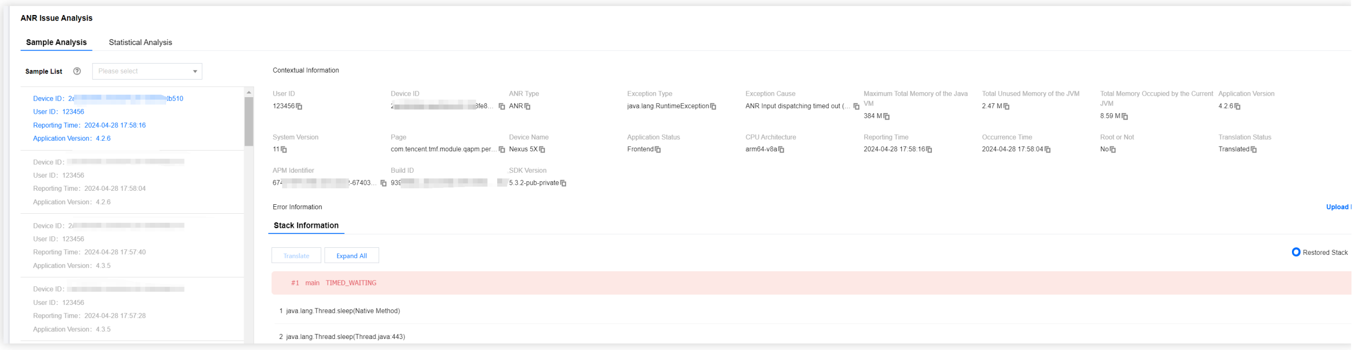
ANR Issue List | All Issue Statuses | Device ID: Please enter the device ID | Specific Function or Filename: Please enter the function

Issue Description	ANR-affected Users (Proportion)	ANR Occurrences (Proportion)	Recent Reporting Time
ID: 2[redacted] java.lang.Thread.sleep(Thread.java) [redacted] com.tencent.tmf.demo.performance.[redacted]File) android.view.View.performClick(View.java)	1 (100%)	4703 (100%)	2024-04-28 17:58:16

Total items: 1 | 10 / page | 1

ANR Issue Details

The details page provides multi-dimensional statistics for a category of issues and analysis of individual cases. You can check the details of an issue by clicking the corresponding issue description.


ANR Issue Analysis

Sample Analysis | Statistical Analysis

Sample List Please select

Device ID: 2[redacted]b510
User ID: 123456
Reporting Time: 2024-04-28 17:58:16
Application Version: 4.2.6

Device ID: [redacted]
User ID: 123456
Reporting Time: 2024-04-28 17:58:04
Application Version: 4.2.6

Device ID: 2[redacted]
User ID: 123456
Reporting Time: 2024-04-28 17:57:40
Application Version: 4.3.5

Device ID: [redacted]
User ID: 123456
Reporting Time: 2024-04-28 17:57:28
Application Version: 4.3.5

Contextual Information

User ID	Device ID	ANR Type	Exception Type	Exception Cause	Maximum Total Memory of the Java VM	Total Unused Memory of the JVM	Total Memory Occupied by the Current JVM	Application Version	
123456	2[redacted]3fe8...	ANR	java.lang.RuntimeException	ANR Input dispatching timed out (...	384 M	2.47 M	8.59 M	4.2.6	
System Version	Page	Device Name	Application Status	CPU Architecture	Reporting Time	Occurrence Time	Root or Not	Translation Status	
11	com.tencent.tmf.module.qapm.per...	Nexus 5X	Frontend	arm64-v8a	2024-04-28 17:58:16	2024-04-28 17:58:04	No	Translated	
APM Identifier	Build ID	SDK Version							
67[redacted]-67403...	93[redacted]	5.3.2-pub-private							

Error Information

Stack Information

Translate | Expand All | Restored Stack

```
#1  main  TIMED_WAITING
1 java.lang.Thread.sleep(Native Method)
2 java.lang.Thread.sleep(Thread.java:443)
```

Metrics Description

ANR-related Metrics are described as follows:

Metric Name	Metrics Description
ANR Rate	Number of devices experiencing ANR/Total number of devices within a specified time range
ANR Times	Number of ANRs occurring in the app within a specified time range
SDK Startups	Number of app launches
ANR-affected User Rate	Number of users affected by ANR/Total number of users launching the app within a specified time range
ANR-affected Users	Number of users affected by ANR within a specified time range
Users	Total number of users launching the app

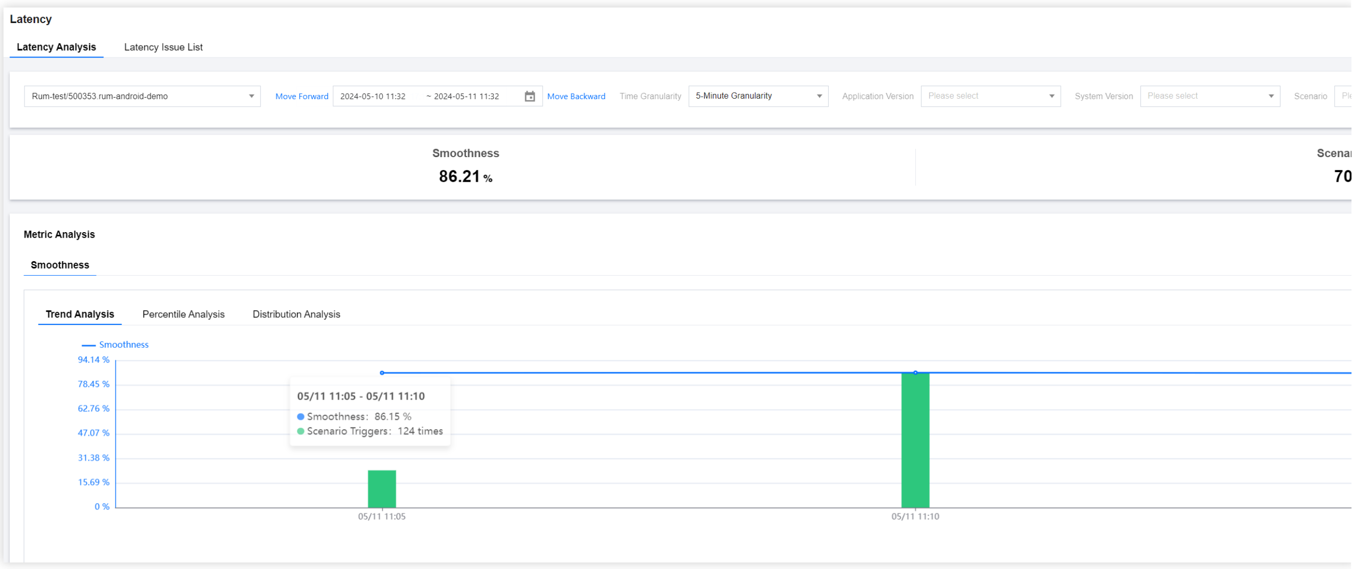
Latency

Last updated：2024-05-20 10:47:31

Mobile App Performance Monitoring aggregates the key features of individual latency cases, facilitating root cause analysis of app latency issues.

Feature Entry

1. Log in to TCOP.
2. In the left menu, click **Mobile App Performance Monitoring** > **Latency**.
3. Select the business system, application, time period, etc., to analyze latency issues.



Rum-test/500353.rum-android-demo
2024-05-10 11:32 ~ 2024-05-11 11:32
Time Granularity 5-Minute Granularity
Application Version Please select
System Version Please select
Scenario

Smoothness 86.21 %
Scenar 70

05/11 11:05 - 05/11 11:10
● Smoothness: 86.15 %
● Scenario Triggers: 124 times

Descriptions

The latency-related metrics descriptions are as follows:

Metric Name	Descriptions
Smoothness	The interval between two frames of a smooth page is 16.67 ms. 1,000 ms (1 second)/16.67 ms = 60 FPS, and 60 FPS represents the ideal smoothness for a page. Only data with a proportion greater than 0.1% is displayed.
Scenario Triggers	Latency Scenario Triggers

Startup

Last updated：2024-05-20 10:47:31

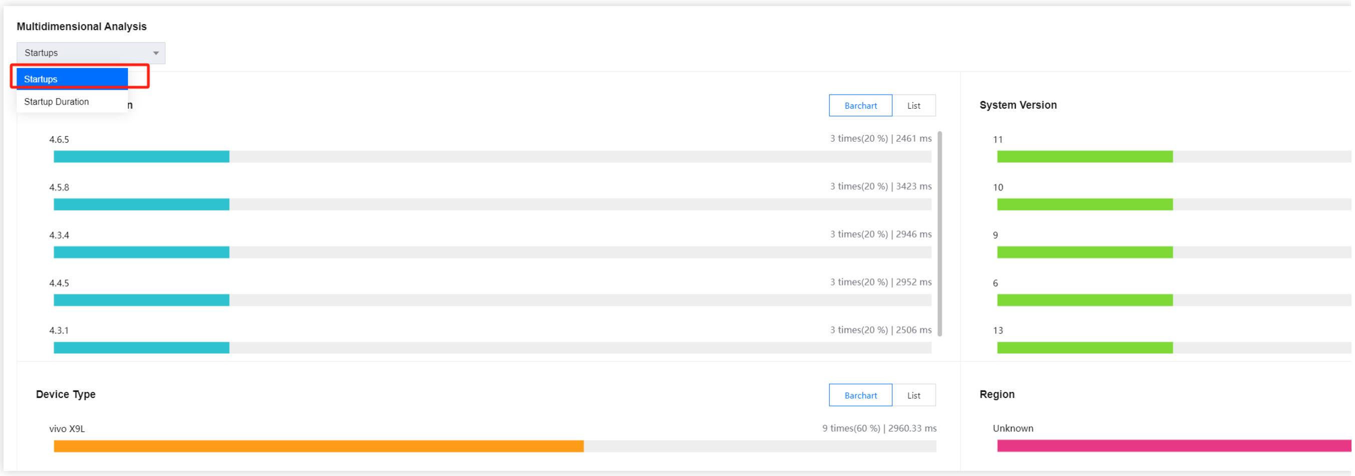
Analyze startup metrics by using Startup Duration and Slow Startup Proportion. You can locate and analyze the root causes of slow app startups through the Slow Startup Issue List.

Feature Entry

1. Log in to Mobile App Performance Monitoring Console.
2. In the left sidebar, select **Startup**, and choose the business system, application, and time range you want to analyze for startup issues.

Multidimensional Analysis

Multidimensional analysis, based on application version, system version, device type, region, ISP, and other dimensions, facilitates targeted analysis of the root causes of observed slow startup.



4.6.5 3 times(20 %) | 2461 ms
4.5.8 3 times(20 %) | 3423 ms
4.3.4 3 times(20 %) | 2946 ms
4.4.5 3 times(20 %) | 2952 ms
4.3.1 3 times(20 %) | 2506 ms

11
10
9
6
13

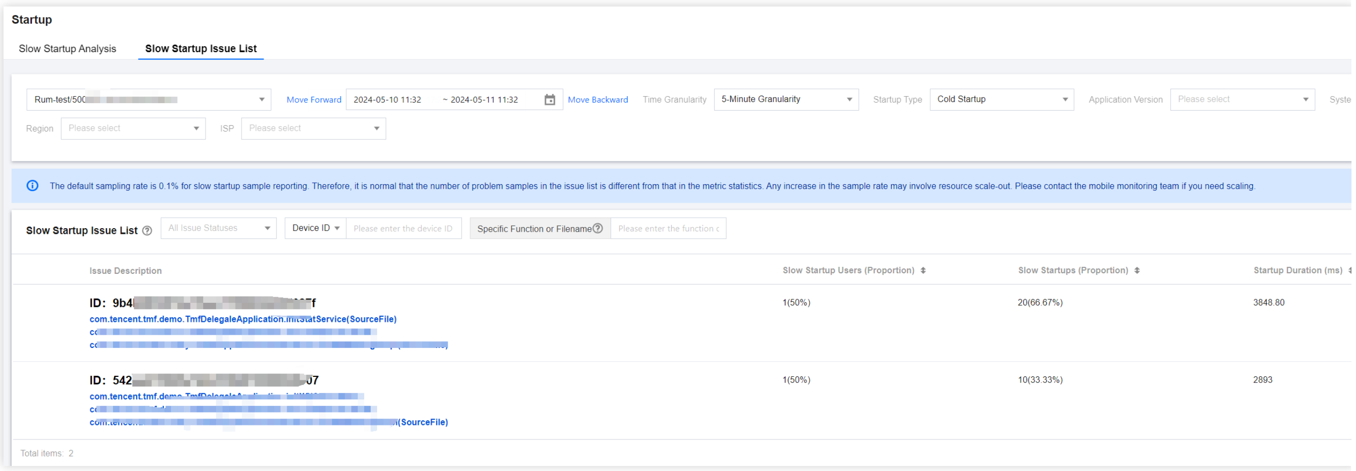
vivo X9L 9 times(60 %) | 2960.33 ms

Unknown

Slow Startup Issue List

The slow startup issue list displays slow startup issues for all devices. You can quickly filter for relevant slow startup devices based on issue exception type, problem device ID, specific function, or file name. You can also click **Issue Description** to view details of the slow startup issue and analyze the root cause of the app's slow startup.

For each application cold/first startup sample, the startup of an Android system with a duration greater than 2,500 ms is considered a slow startup, and the startup of an iOS system with a duration greater than 4,000 ms is considered a slow startup. For each warm startup sample, the startup of an Android system with a duration greater than 1,000 ms is considered a slow startup, and the startup of an iOS system with a startup duration greater than 2,000 ms is considered a slow startup. Only slow startup samples are displayed in the issue list.


Startup

Slow Startup Analysis | **Slow Startup Issue List**

Rum-test/500[redacted] | Move Forward | 2024-05-10 11:32 ~ 2024-05-11 11:32 | Move Backward | Time Granularity: 5-Minute Granularity | Startup Type: Cold Startup | Application Version: Please select | Syste

Region: Please select | ISP: Please select

The default sampling rate is 0.1% for slow startup sample reporting. Therefore, it is normal that the number of problem samples in the issue list is different from that in the metric statistics. Any increase in the sample rate may involve resource scale-out. Please contact the mobile monitoring team if you need scaling.

Slow Startup Issue List | All Issue Statuses | Device ID: Please enter the device ID | Specific Function or Filename: Please enter the function c

Issue Description	Slow Startup Users (Proportion)	Slow Startups (Proportion)	Startup Duration (ms)
ID: 9b4[redacted]**f** com.tencent.tmf.demo.TmfDelegaleApplication.[redacted]tatService(SourceFile) [redacted]	1(50%)	20(66.67%)	3848.80
ID: 542[redacted]**07** com.tencent.tmf.de[redacted] [redacted](SourceFile)	1(50%)	10(33.33%)	2893

Total items: 2

Descriptions

Related metrics descriptions are as follows:

Metric Name	Descriptions
Startups	Application startups.
Startup Duration	Time consumed by the application startup.
Average Duration	The sum of an online sample startup duration/application startups.
Slow Startups (Proportion)	Slow startups/total startups. For an Android terminal, a startup exceeding 2.5 s is, by default, defined as a slow startup. For an iOS terminal, a startup exceeding 4s is, by default, defined as a slow startup, with the threshold being customizable.
Initial Startup	The first startup of the app after installation, considered as a special case of cold startup.
Cold Startup	The process of restarting the app after the app process ended, or running at the backend and the process has been reclaimed by the system.

Warm Startup	The process of awakening the app after it has been in the backend for 3 minutes or switching back from another app's interface.

Network

Last updated：2024-05-13 18:03:25

Network issues are analyzed using metrics such as Throughput, Requests, Network Response Time, Slow Request Proportion, HTTP Error Rate, Network Error Rate, and TCP Connection Establishment Time.

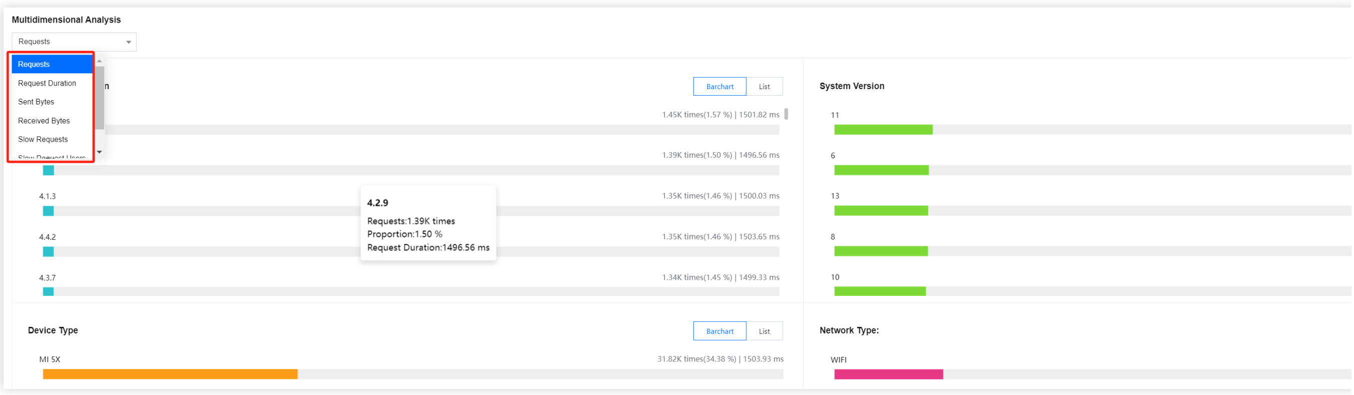
Feature Entry

1. Log in to Tencent Cloud Observability Platform..
2. In the left navigation bar, select **Mobile App Performance Monitoring** > **Network**, You can check network issue analysis from multiple dimensions such as business system, app, and time range.

Multidimensional Analysis

The multidimensional analysis page shows the analysis of key metrics from multiple dimensions such as app version, system version, domain name, URL, device type, network type, region, and internet service provider. It facilitates targeted root cause analysis of specific slow/error requests.

Slow Requests



1.45K times(1.57 %) | 1501.82 ms
1.39K times(1.50 %) | 1496.56 ms
4.1.3 1.35K times(1.46 %) | 1500.03 ms
4.4.2 1.35K times(1.46 %) | 1503.65 ms
4.3.7 1.34K times(1.45 %) | 1499.33 ms

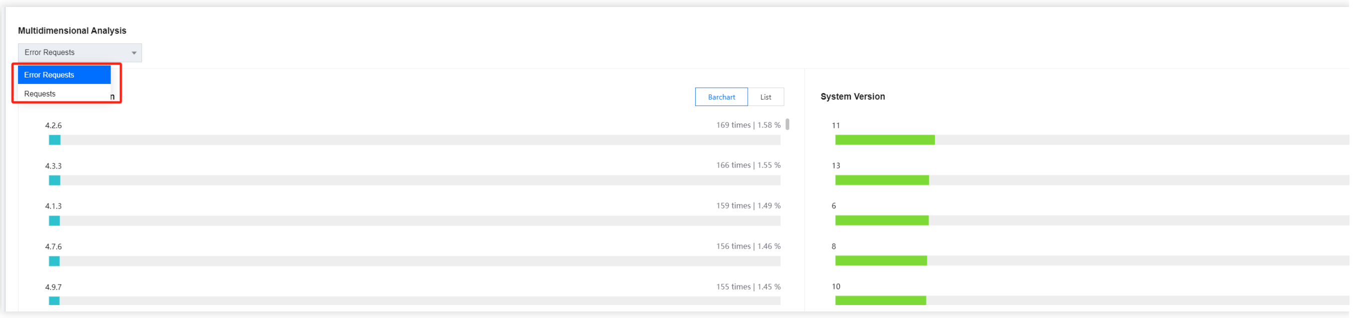
4.2.9
Requests:1.39K times
Proportion:1.50 %
Request Duration:1496.56 ms

System Version: 11, 6, 13, 8, 10

MI 5X 31.82K times(34.38 %) | 1503.93 ms

WIFI

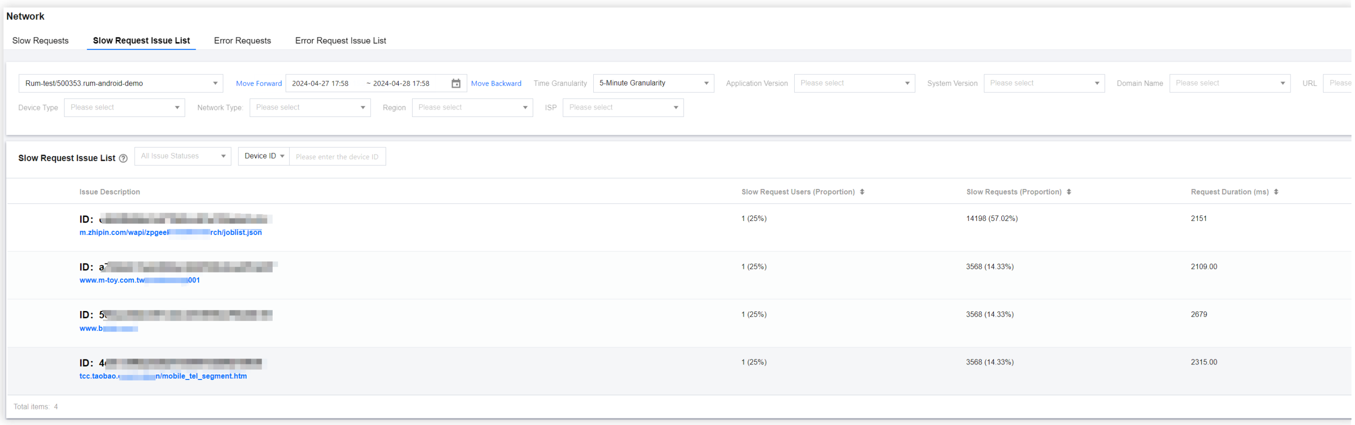
Error Requests



4.2.6	169 times | 1.58 %
4.3.3	166 times | 1.55 %
4.1.3	159 times | 1.49 %
4.7.6	156 times | 1.46 %
4.9.7	155 times | 1.45 %

Slow Request Issue List

The slow request list shows slow requests of all devices. You can quickly filter slow-loading devices by issue type, device ID, specific function, or file name. You can also click the related link under **Issue Description** to view details of slow requests and pinpoint the root cause of slow app requests.



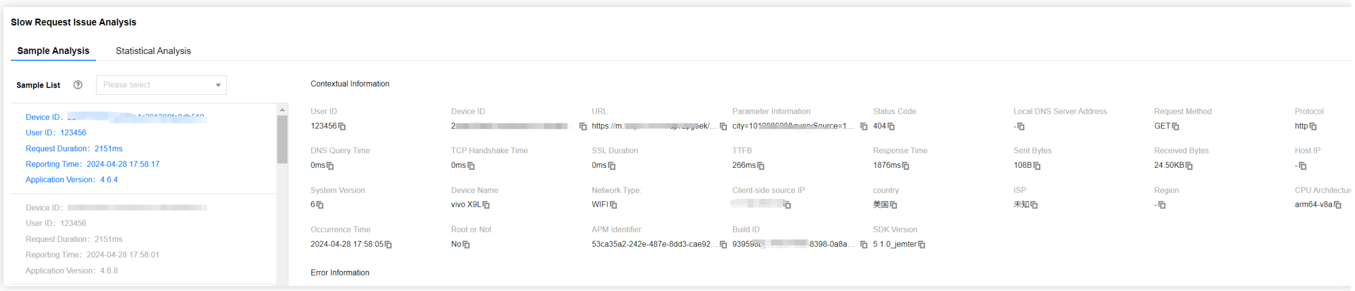
Rum-test/500353.rum-android-demo | 2024-04-27 17:58 ~ 2024-04-28 17:58 | Time Granularity: 5-Minute Granularity | Application Version: Please select | System Version: Please select | Domain Name: Please select | URL: Please
Device Type: Please select | Network Type: Please select | Region: Please select | ISP: Please select

Slow Request Issue List | All Issue Statuses | Device ID | Please enter the device ID

Issue Description	Slow Request Users (Proportion)	Slow Requests (Proportion)	Request Duration (ms)
ID: [redacted] m.zhipin.com/wapi/zpgeel[redacted]rch/joblist.json	1 (25%)	14198 (57.02%)	2151
ID: a[redacted] www.m-toy.com.tw[redacted]001	1 (25%)	3568 (14.33%)	2109.00
ID: 5[redacted] www.b[redacted]	1 (25%)	3568 (14.33%)	2679
ID: 4[redacted] tcc.taobao.[redacted]n/mobile_tel_segment.htm	1 (25%)	3568 (14.33%)	2315.00

Total items: 4

An HTTP request sample is considered a slow request if the transmitted data is over 50 KB and the transfer speed is below 10 KB/s, or if the transmitted data is 50 KB or less and the response time is over 2s. Slow request samples will be shown in the issue list.



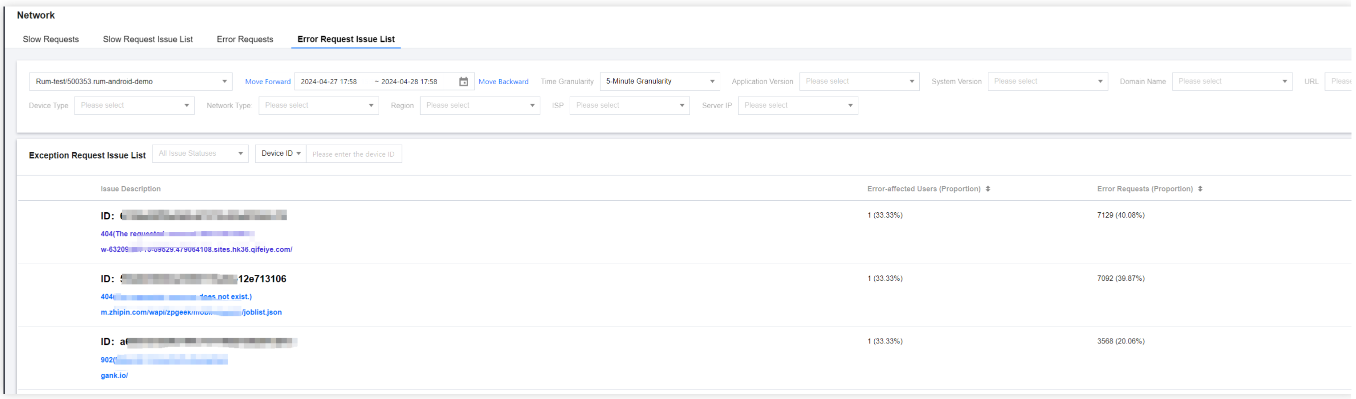
Device ID: [redacted]
User ID: 123456
Request Duration: 2151ms
Reporting Time: 2024-04-28 17:58:17
Application Version: 4.6.4

Device ID: [redacted]
User ID: 123456
Request Duration: 2151ms
Reporting Time: 2024-04-28 17:58:01
Application Version: 4.6.8

User ID	Device ID	URL	Parameter Information	Status Code	Local DNS Server Address	Request Method	Protocol
123456	2[redacted]	https://m.[redacted]ek/...	city=101[redacted]Source=1...	404	-	GET	http
DNS Query Time	TCP Handshake Time	SSL Duration	TTFB	Response Time	Sent Bytes	Received Bytes	Host IP
0ms	0ms	0ms	266ms	1876ms	108B	24.50KB	-
System Version	Device Name	Network Type:	Client-side source IP	country	ISP	Region	CPU Architectur
6	vivo X9L	WIFI	[redacted]	美国	未知	-	arm64-v8a
Occurrence Time	Root or Not	APM Identifier	Build ID	SDK Version			
2024-04-28 17:58:05	No	53ca35a2-242e-487e-8dd3-cae92...	939596[redacted]8398-0a8a...	5.1.0_jemter			

Error Request Issue List

Network errors such as HTTP request errors, DNS resolution errors, failure to establish connection, and connection timeout will be displayed in the error request list. You can click the related link under **Issue Description** to view error request details and pinpoint the root cause of error requests.



Network

Slow Requests | Slow Request Issue List | Error Requests | **Error Request Issue List**

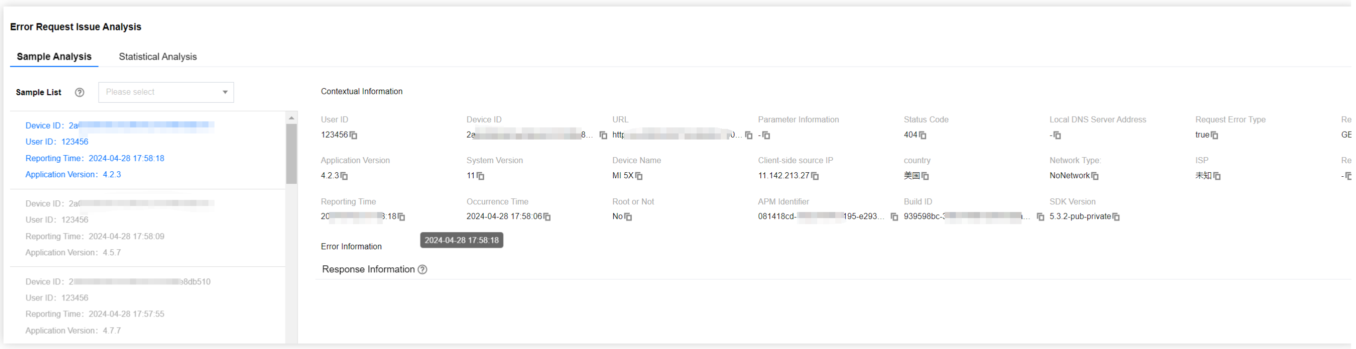
Rum-test/500353.rum-android-demo | Move Forward | 2024-04-27 17:58 ~ 2024-04-28 17:58 | Move Backward | Time Granularity: 5-Minute Granularity | Application Version: Please select | System Version: Please select | Domain Name: Please select | URL: Please

Device Type: Please select | Network Type: Please select | Region: Please select | ISP: Please select | Server IP: Please select

Exception Request Issue List | All Issue Statuses | Device ID | Please enter the device ID

Issue Description	Error-affected Users (Proportion)	Error Requests (Proportion)
ID: [redacted] 404(The requested [redacted] w-63209[redacted].479064108.sites.hk36.qifeiye.com/	1 (33.33%)	7129 (40.08%)
ID: [redacted]12e713106 404([redacted]does not exist.) m.zhipin.com/wapi/zpgeek/[redacted]/joblist.json	1 (33.33%)	7092 (39.87%)
ID: a[redacted] 902([redacted] gank.io/	1 (33.33%)	3568 (20.06%)

You can also click **Issue Description** to view error request details and analyze the cause of errors.



Error Request Issue Analysis

Sample Analysis | Statistical Analysis

Sample List | Please select

Device ID：2a[redacted]
User ID：123456
Reporting Time：2024-04-28 17:58:18
Application Version：4.2.3

Device ID：2a[redacted]
User ID：123456
Reporting Time：2024-04-28 17:58:09
Application Version：4.5.7

Device ID：2[redacted]e8db510
User ID：123456
Reporting Time：2024-04-28 17:57:55
Application Version：4.7.7

Contextual Information

User ID	Device ID	URL	Parameter Information	Status Code	Local DNS Server Address	Request Error Type	Re
123456	2a[redacted]8…	http[redacted]0…	-	404	-	true	GE
Application Version	**System Version**	**Device Name**	**Client-side source IP**	**country**	**Network Type:**	**ISP**	**Re**
4.2.3	11	MI 5X	11.142.213.27	美国	NoNetwork	未知	-
Reporting Time	**Occurrence Time**	**Root or Not**	**APM Identifier**	**Build ID**	**SDK Version**		
20[redacted]:18	2024-04-28 17:58:06	No	081418cd-[redacted]195-e293…	939598bc-3[redacted]…	5.3.2-pub-private		

2024-04-28 17:58:18

Error Information

Response Information

Metrics Description

Related Metrics are as follows:

Metric Name	Metrics Description
Request Duration	App request duration
Slow Request Proportion	The proportion of slow requests to the total number of requests within a selected time range. A request is considered a slow request:

	When the transmitted data is over 50 KB and the transmission speed is below 10 KB/s. When the transmitted data is 50 KB or less and the response time is over 2s.
Slow Requests	The number of slow requests within a selected time range. A request is considered a slow request: When the transmitted data is over 50 KB and the transmission speed is below 10 KB/s. When the transmitted data is 50 KB or less and the response time is over 2s.
Requests	Total application requests
Slow-request Users Proportion	The ratio of the number of users affected by slow requests to the total number of users within a specified time range
Slow Request Users	The number of users affected by slow requests within a specified time range
Request Error Rate	Number of error requests / Total number of requests
Error Requests	Number of network errors in the selected time period Error requests refer to HTTP request errors, DNS resolution errors, inability to establish connection, connection timeout, and other network-related errors
Error-affected User Proportion	Ratio of users affected by error requests to total users within a specified time range
Error-affected Users	Number of users affected by error requests within a specified time range

Webview

Last updated：2024-05-13 18:03:25

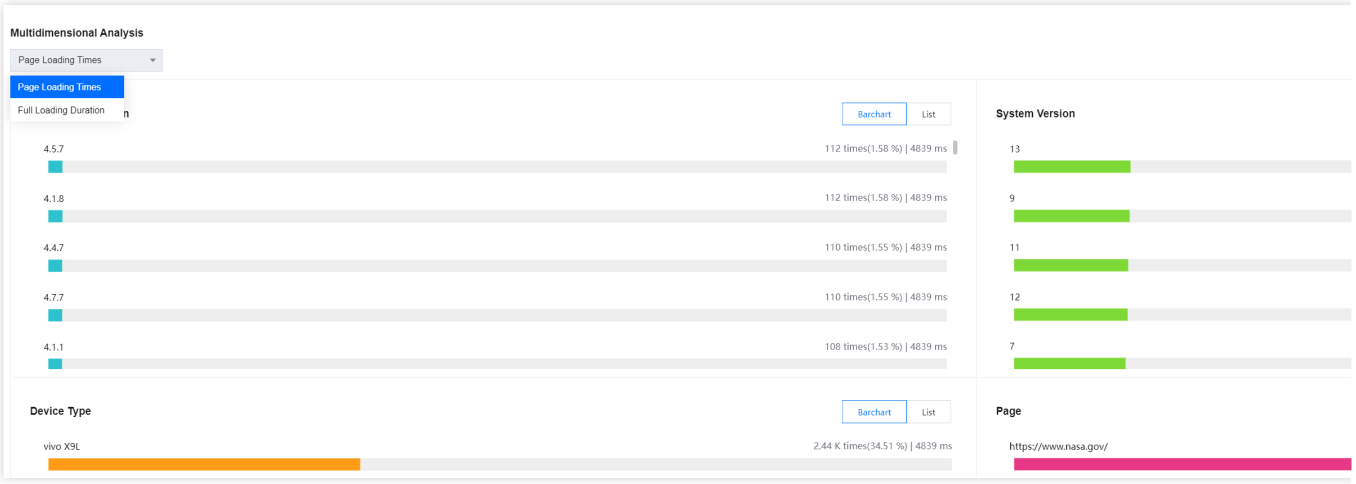
This feature provides WebView metric analysis based on page loading time, slow loading proportion, and JavaScript error rate. It allows you to drill down into WebView and JavaScript errors through the issue list.

Feature Entry

1. log in to Mobile App Performance Monitoring Console.
2. In the left navigation bar, select **WebView**. Select the business system, app, and time range to analyze WebView issues.

Slow loading and JavaScript Error Multidimensional Analysis

The multidimensional analysis page shows the analysis of key metrics from multiple dimensions such as app version, system version, device type, page, network type, internet service provider, and region. It facilitates targeted root cause analysis of specific slow loading issues or JavaScript errors.



4.5.7 112 times(1.58 %) | 4839 ms
4.1.8 112 times(1.58 %) | 4839 ms
4.4.7 110 times(1.55 %) | 4839 ms
4.7.7 110 times(1.55 %) | 4839 ms
4.1.1 108 times(1.53 %) | 4839 ms

13
9
11
12
7

vivo X9L 2.44 K times(34.51 %) | 4839 ms

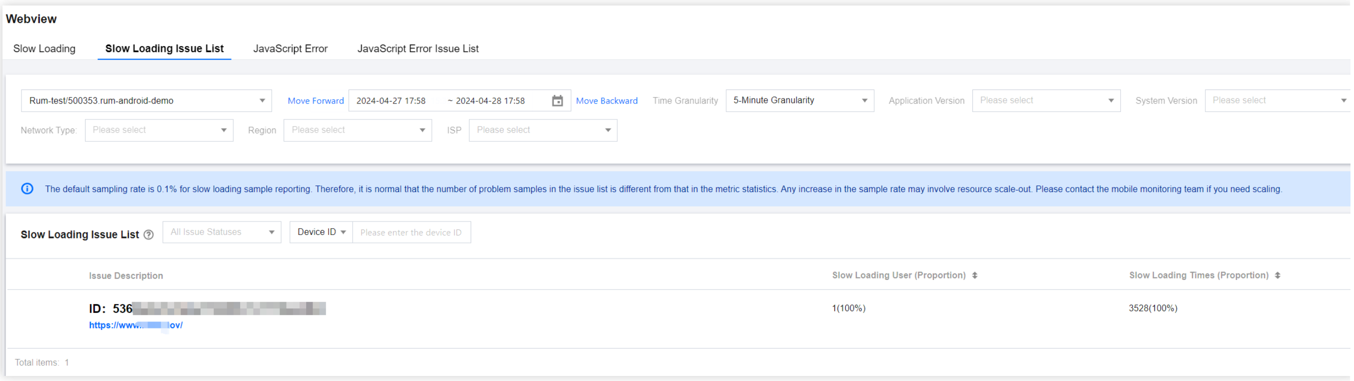
https://www.nasa.gov/

Slow Loading Issues List

The slow loading issue list shows slow loading issues of all devices. You can quickly filter slow-loading devices by error type and device ID. You can also click **Issue Description** to view the details of slow loading and pinpoint and analyze the root causes of slow loading for your app.

Note:

The default sampling rate for slow loading sample reporting is 0.1%, so it is normal for the number of issue samples in the issue list to not match the metric statistics.



Webview

Slow Loading | **Slow Loading Issue List** | JavaScript Error | JavaScript Error Issue List

Rum-test/500353.rum-android-demo | Move Forward | 2024-04-27 17:58 ~ 2024-04-28 17:58 | Move Backward | Time Granularity: 5-Minute Granularity | Application Version: Please select | System Version: Please select

Network Type: Please select | Region: Please select | ISP: Please select

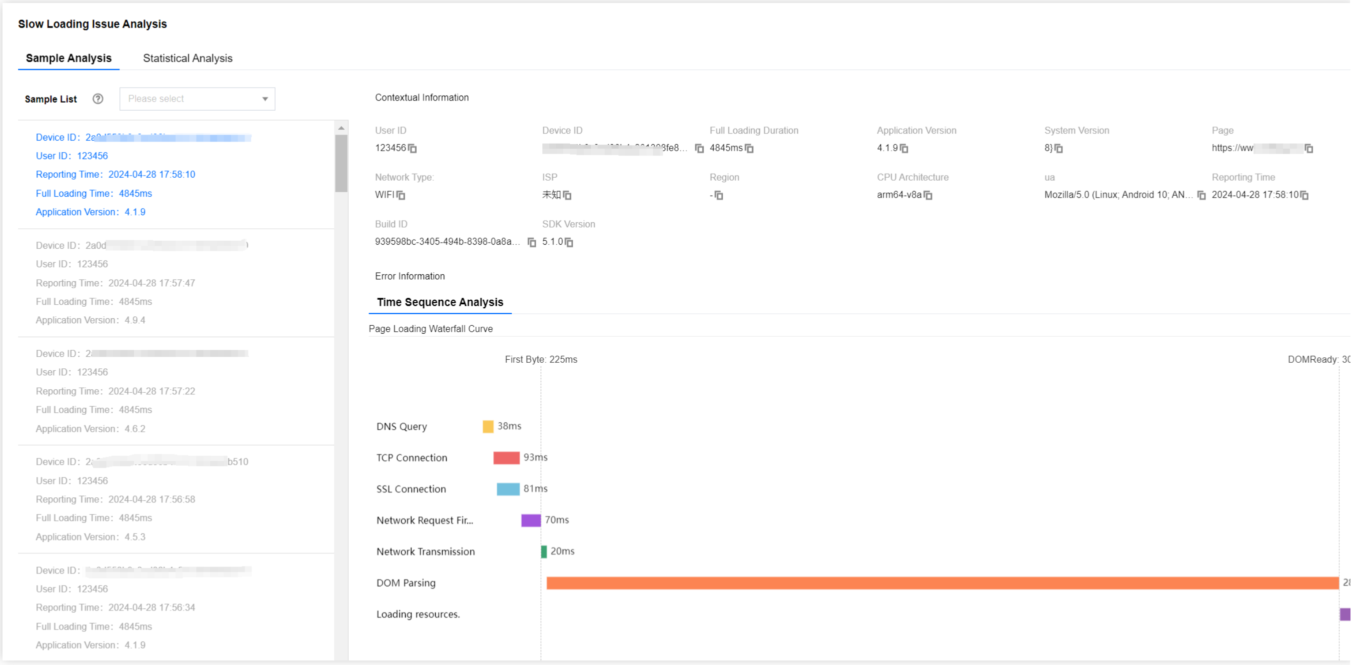
The default sampling rate is 0.1% for slow loading sample reporting. Therefore, it is normal that the number of problem samples in the issue list is different from that in the metric statistics. Any increase in the sample rate may involve resource scale-out. Please contact the mobile monitoring team if you need scaling.

Slow Loading Issue List | All Issue Statuses | Device ID | Please enter the device ID

Issue Description	Slow Loading User (Proportion)	Slow Loading Times (Proportion)
ID: 536[redacted] https://www.[redacted]ov/	1(100%)	3528(100%)

Total items: 1

For each page loading sample, a full loading time greater than 3,500 ms is considered slow loading, and slow loading samples will be displayed in the issue list.



Slow Loading Issue Analysis

Sample Analysis | Statistical Analysis

Sample List | Please select

Device ID: 2a[redacted]
User ID: 123456
Reporting Time: 2024-04-28 17:58:10
Full Loading Time: 4845ms
Application Version: 4.1.9

Device ID: 2a0d[redacted]
User ID: 123456
Reporting Time: 2024-04-28 17:57:47
Full Loading Time: 4845ms
Application Version: 4.9.4

Device ID: 2[redacted]
User ID: 123456
Reporting Time: 2024-04-28 17:57:22
Full Loading Time: 4845ms
Application Version: 4.6.2

Device ID: 2[redacted]b510
User ID: 123456
Reporting Time: 2024-04-28 17:56:58
Full Loading Time: 4845ms
Application Version: 4.5.3

Device ID: [redacted]
User ID: 123456
Reporting Time: 2024-04-28 17:56:34
Full Loading Time: 4845ms
Application Version: 4.1.9

Contextual Information

User ID	Device ID	Full Loading Duration	Application Version	System Version	Page
123456	[redacted]fe8...	4845ms	4.1.9	8}	https://ww[redacted]
Network Type:	**ISP**	**Region**	**CPU Architecture**	**ua**	**Reporting Time**
WIFI	未知	-	arm64-v8a	Mozilla/5.0 (Linux; Android 10; AN...	2024-04-28 17:58:10
Build ID	**SDK Version**				
939598bc-3405-494b-8398-0a8a...	5.1.0				

Error Information

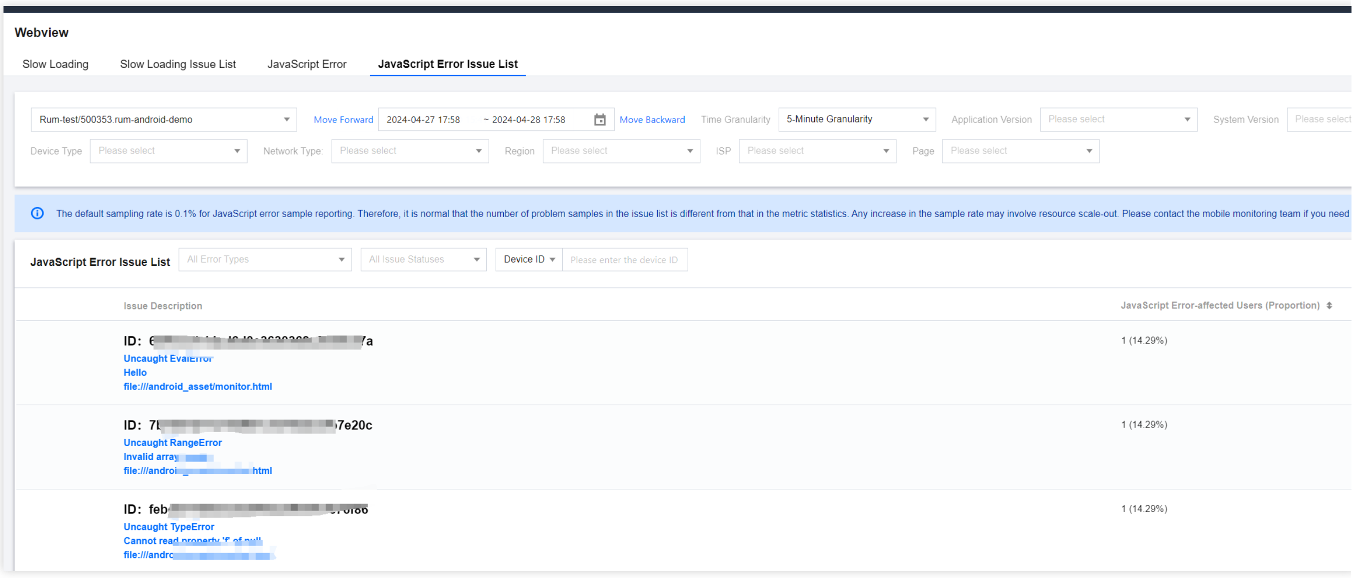
Time Sequence Analysis

Page Loading Waterfall Curve

JavaScript Error Issues List

You can view all JavaScript errors in the JavaScript error Issues list.

Note:
The default sampling rate for JavaScript error reporting is 0.1%, so it is normal for the number of issue samples in the issue list to not match the metric statistics.



Webview

Slow Loading | Slow Loading Issue List | JavaScript Error | **JavaScript Error Issue List**

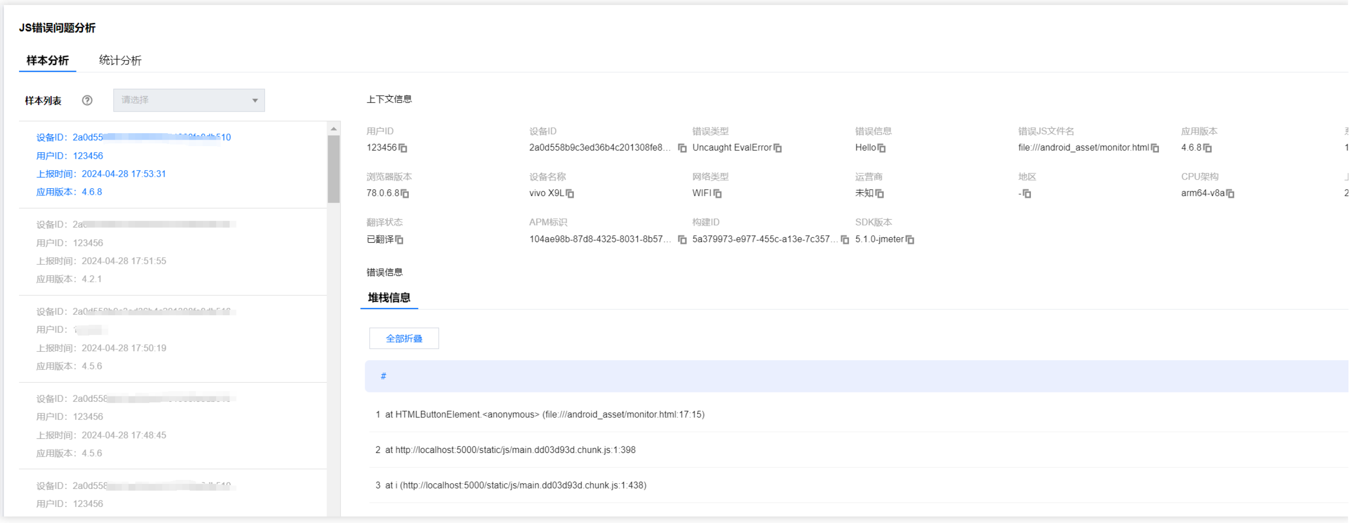
Rum-test/500353.rum-android-demo | Move Forward | 2024-04-27 17:58 ~ 2024-04-28 17:58 | Move Backward | Time Granularity: 5-Minute Granularity | Application Version: Please select | System Version: Please select
Device Type: Please select | Network Type: Please select | Region: Please select | ISP: Please select | Page: Please select

The default sampling rate is 0.1% for JavaScript error sample reporting. Therefore, it is normal that the number of problem samples in the issue list is different from that in the metric statistics. Any increase in the sample rate may involve resource scale-out. Please contact the mobile monitoring team if you need

JavaScript Error Issue List | All Error Types | All Issue Statuses | Device ID: Please enter the device ID

Issue Description	JavaScript Error-affected Users (Proportion)
ID: 6[redacted]7a Uncaught EvalError Hello file:///android_asset/monitor.html	1 (14.29%)
ID: 7b[redacted]7e20c Uncaught RangeError Invalid array[redacted] file:///androi[redacted]html	1 (14.29%)
ID: feb[redacted] Uncaught TypeError Cannot read[redacted] file:///andro[redacted]	1 (14.29%)

You can also click **Issue Description** to view details of JavaScript errors and pinpoint and analyze the causes of JavaScript errors.



JS错误问题分析

样本分析 | 统计分析

样本列表 | 请选择

设备ID：2a0d55[redacted]510
用户ID：123456
上报时间：2024-04-28 17:53:31
应用版本：4.6.8

设备ID：2a[redacted]
用户ID：123456
上报时间：2024-04-28 17:51:55
应用版本：4.2.1

设备ID：2a0[redacted]
用户ID：[redacted]
上报时间：2024-04-28 17:50:19
应用版本：4.5.6

设备ID：2a0d558[redacted]
用户ID：123456
上报时间：2024-04-28 17:48:45
应用版本：4.5.6

设备ID：2a0d558[redacted]
用户ID：123456

上下文信息

用户ID	设备ID	错误类型	错误信息	错误JS文件名	应用版本
123456	2a0d558b9c3ed36b4c201308fe8...	Uncaught EvalError	Hello	file:///android_asset/monitor.html	4.6.8
浏览器版本	**设备名称**	**网络类型**	**运营商**	**地区**	**CPU架构**
78.0.6.8	vivo X9L	WIFI	未知	-	arm64-v8a
翻译状态	**APM标识**	**构建ID**	**SDK版本**		
已翻译	104ae98b-87d8-4325-8031-8b57...	5a379973-e977-455c-a13e-7c357...	5.1.0-jmeter		

错误信息

堆栈信息

全部折叠

```
#
1 at HTMLButtonElement.<anonymous> (file:///android_asset/monitor.html:17:15)
2 at http://localhost:5000/static/js/main.dd03d93d.chunk.js:1:398
3 at i (http://localhost:5000/static/js/main.dd03d93d.chunk.js:1:438)
```

Metrics Description

Related Metrics are as follows:

Metric Name	Metrics Description
Page Loading Times	Number of times a page is opened or refreshed
Full Loading Duration	Time taken for the entire web page to be fully loaded
JavaScript Errors	Total number of JavaScript errors within a specified time range
JavaScript Error Rate	Number of users experiencing JavaScript errors/Total number of users accessing the WebView page. Due to computational resource limitations, the numerator and denominator of this metric are not deduplicated.
JavaScript Error-affected User Proportion	Number of users affected by JavaScript errors/Total number of users within a specified time range
JavaScript Error-affected Users	Number of users affected by JavaScript errors within a specified time range

Application Management

Last updated：2024-05-13 18:03:25

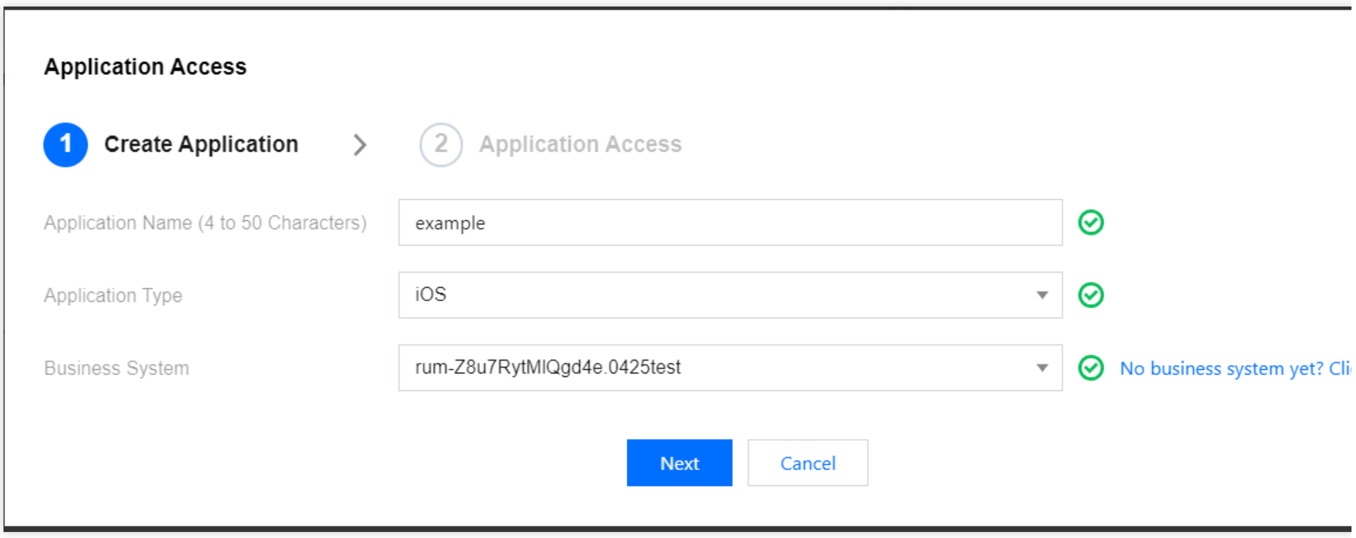
Use Cases

On the app management page, you can view your connected end-user apps, new app access, and app IDs, and set an allowlist.

Directions

Accessing App

1. Log in to Tencent Cloud Observability Platform.
2. In the left sidebar, select **Mobile App Performance Monitoring** > **Overview**.
3. click **Application Access**, fill in the app name, set the app type to Android or iOS, set the business system, and click **Next**.



Application Access

1 Create Application > 2 Application Access

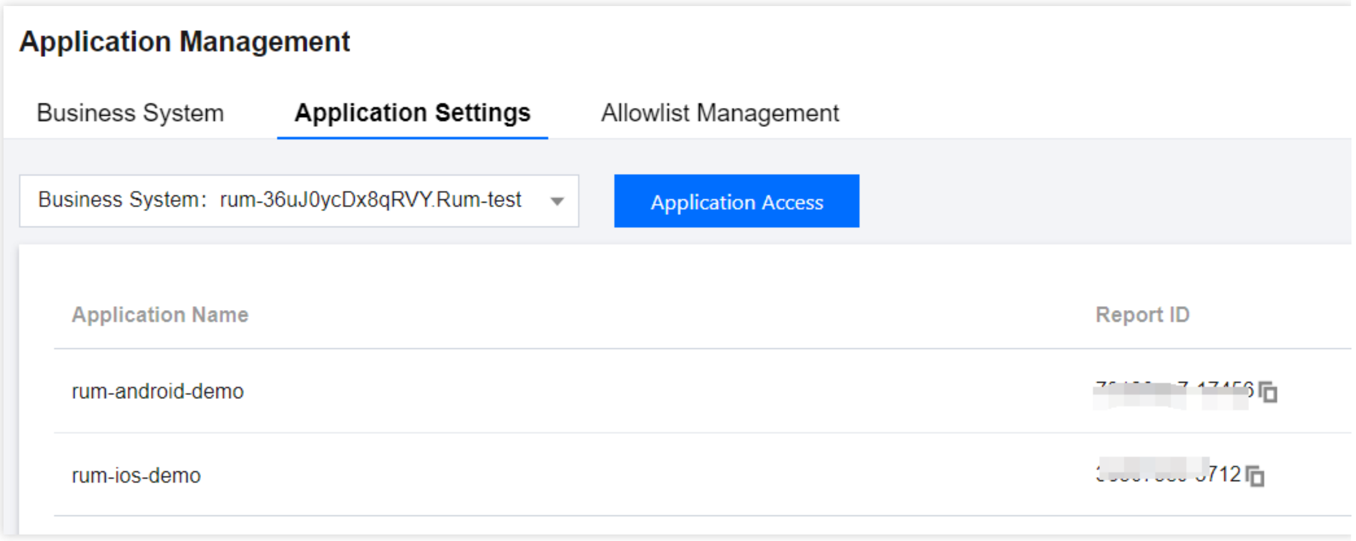
Application Name (4 to 50 Characters): example

Application Type: iOS

Business System: rum-Z8u7RytMlQgd4e.0425test

Obtaining App ID

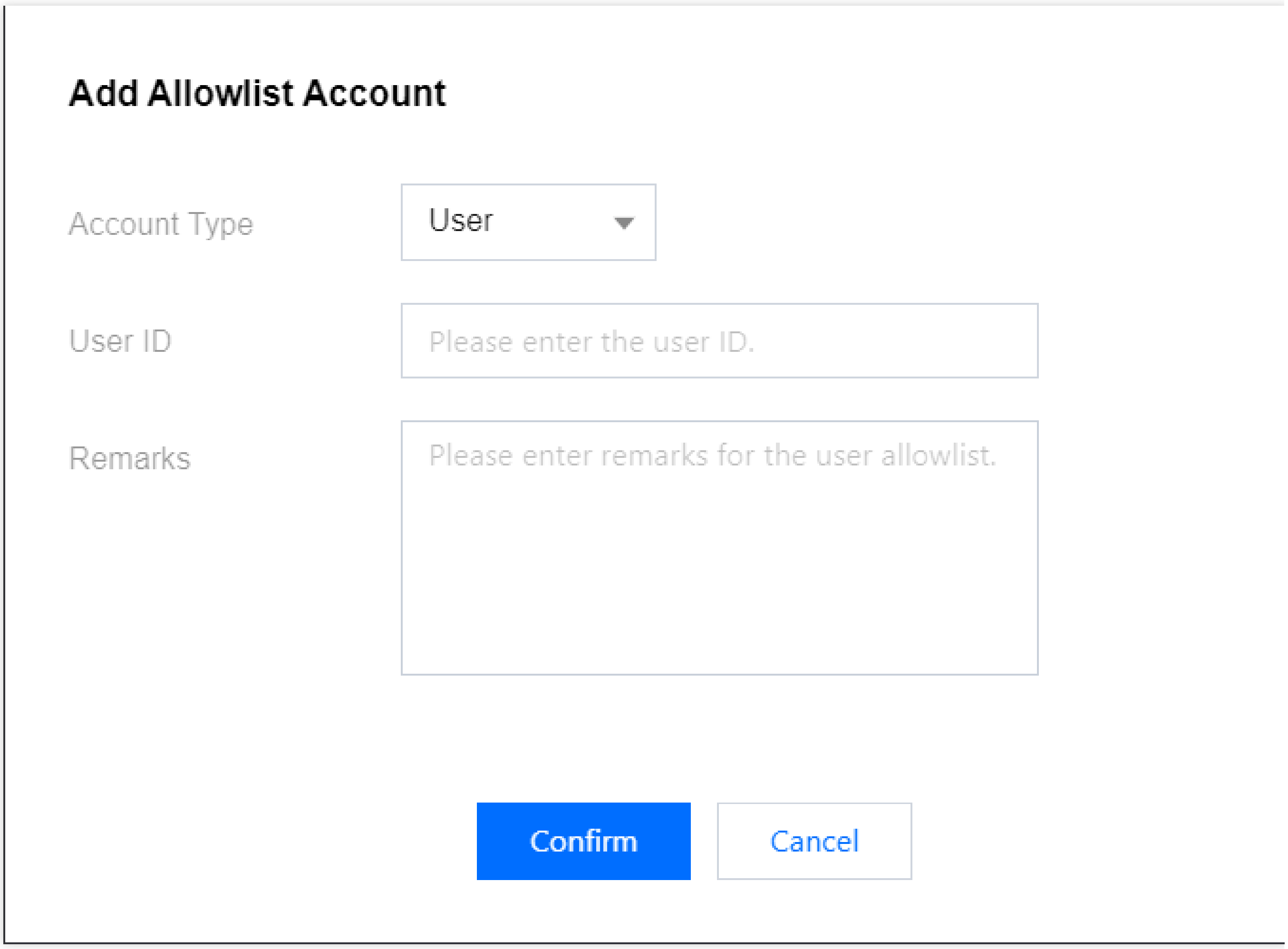
On the **Application Management** > **Application Settings** to enter the app settings page and obtain app IDs from the list.



Application Name	Report ID
rum-android-demo	[redacted]
rum-ios-demo	[redacted]

Allowlist Configuration

Select **Application Management > Allowlist Management** to enter the allowlist configuration page. Click **Add** on the allowlist configuration page. Configure a user ID/device ID allowlist to prevent data reporting from being affected by sampling. That is, users/devices in the allowlist are not affected by the sampling rate and will all be sampled. You can select a user or device type and fill in the relevant ID.


Add Allowlist Account

Account Type	User
User ID	Please enter the user ID.
Remarks	Please enter remarks for the user allowlist.