

CC直播对接问题解决方案汇总

版本：3.2.1 日期：2019-02-28

1. 文档介绍

1.1 文档目的

旨在帮助客户更好的使用CC直播Android SDK，快速解决对接开发过程中可能遇到的一些问题。

2. 常见问题

2.1 登陆直播间失败

解决方式：

1. 检查AndroidManifest.xml是否有android.permission.INTERNET权限
2. 检查DWLive.getInstance().setDWLiveLoginParams(...);设置的参数是否正确

2.2 直播间登陆成功了，跳转到播放界面的时候，一直在转圈，没有出现画面

解决方式：查看ioSocket版本，目前SDK不支持0.9.0版本及以上版本。如果已经有okhttp的库了，并存在一定的版本冲突，则推荐使用0.8.3版本。

2.3 直播间登录操作和播放操作做在一起了，为什么直播登录成功后，没有视频加载出来

解决方式：建议将登录和播放操作分为两个页面。因为直播、回放的相关数据需要在播放之前获取到，而播放又跟surface的生命周期有关系。

3. 常见集成SDK时会出现的Crash

注：对比日志信息的时候，行号不需要太在意，内容能匹配基本上就可以定位当前存在的问题。

3.1 集成SDK时，出现崩溃，日志为：

```
FATAL EXCEPTION: main
Process: com.bokecc.dwlivedemo_new, PID: 12242
java.lang.NoClassDefFoundError: com.bokecc.sdk.mobile.live.DWLive$28
    at com.bokecc.sdk.mobile.live.DWLive.<init>
(DWLive.java:903)
    at com.bokecc.sdk.mobile.live.DWLive.<clinit>
(DWLive.java:116)
    .....省略
    at android.view.View$PerformClick.run(View.java:21350)
    at android.os.Handler.handleCallback(Handler.java:739)
    at android.os.Handler.dispatchMessage(Handler.java:95)
    at android.os.Looper.loop(Looper.java:148)
    at
android.app.ActivityThread.main(ActivityThread.java:5582)
    at java.lang.reflect.Method.invoke(Native Method)
    .....省略
```

解决方案:

检查build.gradle的dependencies里面是否有:

```
compile('io.socket:socket.io-client:0.7.0') {
    exclude group: 'org.json', module: 'json'
}
```

如果没有就加上。

3.2 集成SDK时，出现崩溃，日志为：

```
java.lang.NoClassDefFoundError: Failed resolution of:
Lorg/webrtc/PeerConnection$Observer;
    at com.bokecc.sdk.mobile.live.DWLive.b(DWLive.java:498)
    at com.bokecc.sdk.mobile.live.DWLive.d(DWLive.java:71)
    at com.bokecc.sdk.mobile.live.DWLive$4.run(DWLive.java:391)
    at java.lang.Thread.run(Thread.java:761)
Caused by: java.lang.ClassNotFoundException: Didn't find class
"org.webrtc.PeerConnection$Observer" on path: DexPathList[[zip file
"/data/app/cn.btclass.sierra-2/base.apk"],
nativeLibraryDirectories=[data/app/cn.btclass.sierra-2/lib/arm64,
/system/fake-libs64, /data/app/cn.btclass.sierra-2/base.apk!/lib/arm64-v8a,
/system/lib64, /vendor/lib64]]
```

解决方案:

引入 libjingle_peerconnection.jar 包

3.3 集成SDK时，出现崩溃，日志为：

```

java.lang.UnsatisfiedLinkError: dalvik.system.PathClassLoader[DexPathList[[zip
file "/data/app/com.bokecc.dwlivedemo_new-2/base.apk", zip file
"/data/app/com.bokecc.dwlivedemo_new-2/split_lib_dependencies_apk.apk", zip
file "/data/app/com.bokecc.dwlivedemo_new-2/split_lib_slice_0_apk.apk", zip
file
....省略若干日志
/system/vendor/lib64, /product/lib64]]] couldn't find "libijkffmpeg.so"
  at java.lang.Runtime.loadLibrary0(Runtime.java:989)
  at java.lang.System.loadLibrary(System.java:1533)
  at
tv.danmaku.ijk.media.player.IjkMediaPlayer$1.loadLibrary(IjkMediaPlayer.java:1
75)
  at
tv.danmaku.ijk.media.player.IjkMediaPlayer.loadLibrariesOnce(IjkMediaPlayer.ja
va:186)
  at
tv.danmaku.ijk.media.player.IjkMediaPlayer.initPlayer(IjkMediaPlayer.java:228)
  at tv.danmaku.ijk.media.player.IjkMediaPlayer.<init>(IjkMediaPlayer.java:224)
  at tv.danmaku.ijk.media.player.IjkMediaPlayer.<init>(IjkMediaPlayer.java:215)
  at
com.bokecc.dwlivedemo_new.activity.ReplayActivity.initPlayer(ReplayActivity.ja
va:177)
  at
com.bokecc.dwlivedemo_new.activity.ReplayActivity.onCreate(ReplayActivity
.java:154)
  at com.bokecc.dwlivedemo_new.base.BaseActivity.onCreate(BaseActivity.java:35)
  at android.app.Activity.performCreate(Activity.java:6910)
  at
android.app.Instrumentation.callActivityOnCreate(Instrumentation.java:1123)
  at android.app.ActivityThread.performLaunchActivity(ActivityThread.java:2746)
  at android.app.ActivityThread.handleLaunchActivity(ActivityThread.java:2864)
  at android.app.ActivityThread.-wrap12(ActivityThread.java)
  at android.app.ActivityThread$H.handleMessage(ActivityThread.java:1567)
  at android.os.Handler.dispatchMessage(Handler.java:105)
  at android.os.Looper.loop(Looper.java:156)
  at android.app.ActivityThread.main(ActivityThread.java:6523)
  at java.lang.reflect.Method.invoke(Native Method)
  at
com.android.internal.os.ZygoteInit$MethodAndArgsCaller.run(ZygoteInit.java:941
)
  at com.android.internal.os.ZygoteInit.main(ZygoteInit.java:831)

```

解决方案：

检查是否引入了SDK提供的jniLibs文件夹里面的so库。

3.4 集成SDK时，出现崩溃，日志为：

```
java.lang.NoClassDefFoundError:com.bokecc.sdk.mobile.live.rtc.RtcClient$1
    at com.bokecc.sdk.mobile.live.rtc.RtcClient.<init>(RtcClient.java:101)
    .... 省略
```

解决方案：

一般是集成了有连麦功能的SDK的jar包，但是又不需要，没引入libjingle_peerconnection.jar，导致的，解决方案：要么用户引入libjingle_peerconnection.jar包，要么用无连麦的jar包，根据用户实际情况看。

3.5 集成SDK时出现崩溃，日志为：

```
Caused by: java.lang.NullPointerException: Attempt to invoke virtual method
'java.lang.String com.bokecc.sdk.mobile.live.pojo.TemplateInfo.getPdfView()'
on a null object reference
    at com.bokecc.sdk.mobile.live.DWLive.setDWLivePlayParams(DWLive.java:212)
```

解决方案：一般情况下用户把登录操作方法startLogin()和设置播放相关参数的setDWLivePlayParams()写在一起了，大概率用户写的代码会是下面的形式：

```
DWLive.getInstance().setDWLiveLoginParams(****);
DWLive.getInstance().startLogin();
// 紧接着就写下面的代码了，这样写是错误的，正确的应该是监听到登录成功后再执行下面的方法
DWLive.getInstance().setDWLivePlayParams(****);
```

解决方式就是上面代码的注释部分。

3.6 如果集成SDK时出现类似如下的日志

```
System.err: java.lang.ClassNotFoundException: Didn't find class
"org.webrtc.MediaCodecVideoEncoder$OutputBufferInfo" on path: DexPathList[[zip
file "/data/app/com.allin.social-2/base.apk"],nativeLibraryDirectories=
[/data/app/com.allin.social-2/lib/arm, /data/app/com.allin.social-
2/base.apk!/lib/armeabi, /vendor/lib, /system/lib]]
.....
System.err: Caused by: java.lang.NoClassDefFoundError: Class not found using
the boot class loader; no stack trace available
E/rtc: #
    # Fatal error in ../../webrtc/api/android/jni/classreferenceholder.cc,
line 122
    # last system error: 2
    # Check failed: !jni->ExceptionCheck()
    # error during FindClass:
org/webrtc/MediaCodecVideoEncoder$OutputBufferInfo
    #
```

检查混淆配置是否如下：

```
-keep public class com.bokecc.sdk.mobile.**{*;}
-keep public class tv.**{*;}
-keep public interface com.bokecc.sdk.mobile.**{*;}
-keep public class org.webrtc.**{*;}
-keep class com.intel.webrtc.base.**{*;}
-keep interface com.intel.webrtc.base.**{*};}
```

3.6 在线回放或者离线回放OOM 问题

使用SDK播放在线回放或者离线回放的时候，如果出现OOM，可以将问题反馈给客服，然后将相关回放链接信息提供出来，由SDK的研发人员查看本回放的相关数据情况。

客户可以通过设置应用的大内存模式来尝试解决：

```
<application
    android:name=".DWApplication"
    android:largeHeap="true"    // 设置大内存模式
    android:theme="@style/AppTheme">
```

4. 可能需要的特性

4.1 视频和文档区域交换父容器：

切换的时候基本的思路是：

```
rlLiveTopLayout.removeView(mPlayerContainer); // 从原父容器移除（先）
rlLiveInfosLayout.addView(mPlayerContainer); // 添加到新的父容器（后）
```

上面那个是视频移出移入的方式。文档区域同理。但是仅仅这样做，每次切换都会重新加载，为了防止重新加载，需要控制surface的生命周期里面的一些逻辑：

```
@Override
public void onSurfaceTextureAvailable(SurfaceTexture surfaceTexture, int i, int i1) {
    surface = new Surface(surfaceTexture);
    if (isOnResumeStart) {
        return;
    }
    if (player.isPlaying()) {
        player.setSurface(surface);
    } else {
        dwLive.start(surface);
    }
}
```

这块的特性，目前在Demo已经做了支持，建议参考Demo实现。

5. SDK兼容的问题的建议

5.1 okhttp 和 io.socket 版本建议:

```
implementation 'com.squareup.okhttp3:okhttp:3.10.0'
implementation('io.socket:socket.io-client:0.8.3') {
    exclude group: 'org.json', module: 'json'
}
```

由于需要和服务端兼容，目前io.sokcet版本最高就是0.8.3。

5.2 Android 9.0 兼容的建议:

目前SDK的数据请求绝大部分都使用了Https的方式，但是还是有少数的Http请求。

为更好的适配Android 9.0，建议在AndroidManifest.xml中的Application层添加配置信息：

```
android:networkSecurityConfig="@xml/network_security_config"
```

然后在res的xml文件夹中添加：network_security_config.xml

```
<?xml version="1.0" encoding="utf-8"?>
<network-security-config>
    <base-config cleartextTrafficPermitted="true" />
</network-security-config>
```

6. 对接问题反馈建议

为了加快对接问题的处理速度，那么反馈问题的时候，请按照以下思路进行反馈：

1. 反馈的问题是"点播/直播/小班课"的问题，这个问题是"Andorid/iOS/小程序的问题，是"客户端/网页端"的问题
2. 如果是云直播Android端的问题，说明下使用的是直播观看/在线回放/离线回放，并提供复现步骤及相关日志

(已经到底部)*