

# 黑鲨 SDK 联运接入常见问题 FAQ

## Q: 用什么手机调试

A: 可以使用黑鲨手机或其他 Android 手机调试，Android 版本推荐 9.0 及以上。

## Q: 初始化失败

A: 确保在 AndroidManifest.xml 声明了 SDK 依赖的所有组件。

```
<!--不支持${applicationId}的请替换为包名-->
<provider
    android:name="com.blackshark.gamecenter.sdk.provider.BsFileProvider"
    android:authorities="${applicationId}.shark_provider"
    android:exported="false"
    android:grantUriPermissions="true">

    <meta-data
        android:name="android.support.FILE_PROVIDER_PATHS"
        android:resource="@xml/provider_paths" />

</provider>
```

## Q: 非黑鲨手机，登录界面或悬浮窗下载”游戏服务”安装失败

A: 检查 AndroidManifest.xml 中 provider

## Q: 点击登录按钮应用 crash

A: (1) 请对照集成文档确认集成步骤都已完成，如果使用 eclipse 集成，请确认相关 jar 包都已添加引用。

(2) 已安装黑鲨游戏服务，需给与自动启动以及悬浮窗权限。

## Q: 安装游戏服务 App 后，点击登录没反应

A: 进入设置-应用管理，点击显示所有应用，搜索黑鲨游戏服务 app，打开自启动权限后再重试。

## Q: 安装游戏服务 App 后回到游戏界面没有继续登录

A: 调登录接口后安装游戏服务 App 回到游戏界面时，需要提供登录按钮给用户点击登录，重新触发登录流程。

## Q: 登录成功但不显示悬浮窗

A: 如果接入的是游戏 SDK 4.0.0 以上版本，登录成功后都会有悬浮窗显示，无论是否安装游戏服务。如果接入

的是游戏 SDK 4.0.0 版本及以上，能够登录成功，但不显示悬浮窗，请检查：

(1) `onUserAgreed` 是否已在游戏主 Activity 的 `onCreate`方法中调用

(2) 确保 `init` 接口是在所有 SDK 的方法调用之前调用并初始化成功

**Q：登录成功后，点击悬浮窗崩溃。**

A：(1) 检查 `res`、`libs` 和 `android support` 依赖是否严格按照文档说明接入；

(2) 检查 R 文件是否被混淆，确保 R 文件及其内部类和类成员不被混淆。

**Q：游戏支付时，切到桌面再回来，支付界面消失，无法继续支付**

A：把游戏启动 activity 的 `launchMode` 设成 `standard`。

若游戏启动 activity 同时是主 activity，不能修改 `launchMode`，建议增加一个 `launchMode=standard` 的闪屏 activity 作为启动入口。

**Q：登录验证返回 `errcode=2` 是什么意思？**

A：一般为登录签名错误，请确认以下事项：

(1) 待加签字符串的顺序和 `key` 名是否正确，正确格式为：

`appId={appId}&session={session}&uid={uid}&app_key={appKey}`。

(2) 使用的密钥是否为 `appKey`。

(3) `session` 参数是否为最近一次登录返回的 `session` 值。

(4) 登录的 `appId` 和 `session` 验证的 `appId` 是否一致。

**Q：登录验证返回 `field appId is not set` 是什么意思？**

A：检查请求参数，`appId` 的 I 是大写。

**Q：支付提示支付失败**

A：请确认相关参数都已按照接入文档要求传入，并检查参数签名方式是否正确，必须和 `demo` 中的签名方式保持一致。必要时可直接把 `demo` 中的签名代码复制到项目中使用。

**Q：支付提示无法获取游戏信息或悬浮球菜单无内容，请先到开放平台发布游戏**

A：联系黑鲨运营同学进行发布



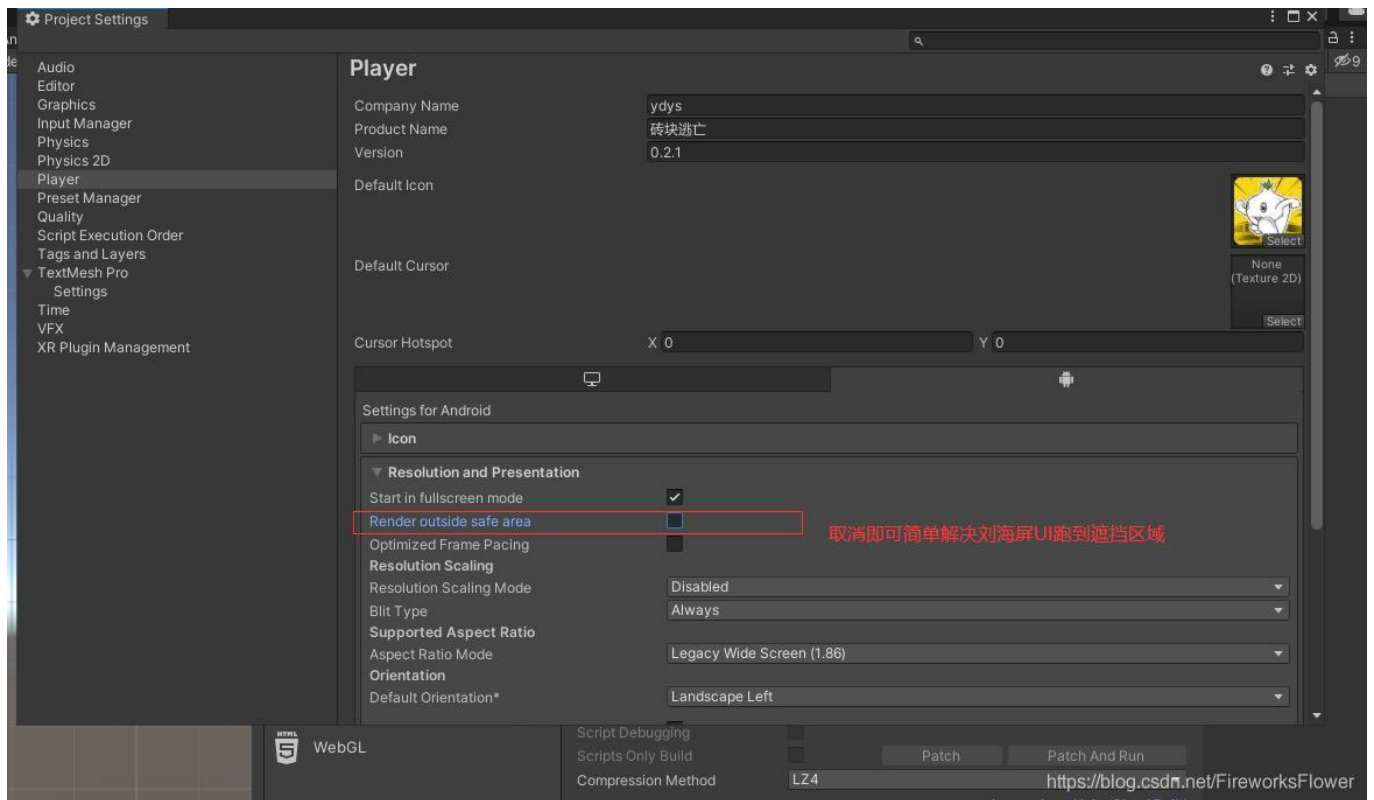
**Q: 游戏运行崩溃，提示缺少资源或依赖**

A: 检查 res、assets、libs 等依赖是否严格按照文档说明接入。尤其注意 eclipse 的 support 接入方式。

若使用gradle 编译，推荐使用 implementation 'com.android.support:support-v4:27.1.1' 进行support 库的依赖。

**Q: Activity 已适配全面屏，游戏主画面仍然存在黑边**

A: 如果 CP 使用 Unity 请确认是否勾选 Render outside safe area



### Q: Unity 4 悬浮球图标无法点击

A: 请在继承了 UnityPlayerNativeActivity 的标签中添加如下配置：

```
<activity android:name="com.unity3d.player.UnityPlayerNativeActivity"
    android:label="@string/app_name">
    ...其它配置
    <meta-data android:name="unityplayer.UnityActivity" android:value="true" />
    <meta-data android:name="unityplayer.ForwardNativeEventsToDalvik"
android:value="true" />
</activity>
```

### Q: Unity 5.6.x 无法显示悬浮球

A: 自定义 UnityPlayer，重写 addView 方法，将 SurfaceView 的 zOrderOnTop 设为 false。示例代码如下

```

/*****自定义 UnityPlayerActivity*****/
package com.xxx.yyy;

import com.unity3d.player.*;
import android.os.Bundle;

public class CUnityPlayerActivity extends UnityPlayerActivity {
    @Override
    public void onCreate(Bundle bundle) {
        requestWindowFeature(1);
        super.onCreate(bundle);
        getWindow().setFormat(2);
        //mUnityPlayer = new UnityPlayer(this);
        mUnityPlayer = new CUnityPlayer(this);
        setContentView(mUnityPlayer);
        mUnityPlayer.requestFocus();
    }

/*****自定义 UnityPlayer*****/
package com.xxx.yyy;

import com.unity3d.player.*;
import android.content.ContextWrapper;
import android.view.SurfaceView;
import android.view.View;

public class CUnityPlayer extends UnityPlayer {
    public CUnityPlayer(ContextWrapper contextwrapper) {
        super(contextwrapper);
    }

    public void addView(View child) {
        if (child instanceof SurfaceView) {
            ((SurfaceView)child).setZOrderOnTop(false);
        }
        super.addView(child);
    }
}

```

**Q: 登录时返回错误码 100000**

A: 检查请求参数 `appId` 是否正确。

**Q: 包名发生变化，后台重新修改包名后，发现参数有变化**

A: 将项目中使用的对应参数替换为新的参数。

**Q: 手机深色模式下，支付宝界面输入不了密码**

A: 进入手机设置界面，搜索深色模式，关闭系统的深色模式。

**Q: 按下返回键后，先退出到桌面，再弹出退出弹窗**

A: 请确认是否拦截了返回键事件，可以参考 [demo](#) 示例。

**Q: 打孔屏黑边问题**

A: 适配打孔屏: 在游戏的主 activity 的 onCreate() 中加入

```
if (Build.VERSION.SDK_INT >= 28) {  
    mActivity.getWindow().getDecorView().setSystemUiVisibility(  
        View.SYSTEM_UI_FLAG_FULLSCREEN  
        | View.SYSTEM_UI_FLAG_LAYOUT_FULLSCREEN);  
    WindowManager.LayoutParams lp =  
        mActivity.getWindow().getAttributes();  
    lp.layoutInDisplayCutoutMode =  
        WindowManager.LayoutParams  
            .LAYOUT_IN_DISPLAY_CUTOUT_MODE_SHORT_EDGES;  
    mActivity.getWindow().setAttributes(lp);  
}
```