**百度地图：**

一、注意事项

1、静态库中采用ObjectC++实现，因此需要您保证您工程中至少有一个.mm后缀的源文件(您可以将任意一个.m后缀的文件改名为.mm)，或者在工程属性中指定编译方式，即将Xcode的Project -> Edit Active Target -> Build -> GCC4.2 - Language -> Compile Sources As设置为"Objective-C++"

2、由于iOS9改用更安全的https，为了能够在iOS9中正常使用地图SDK，请在"Info.plist"中进行如下配置，否则影响SDK的使用。

<key>NSAppTransportSecurity</key>

<dict>

<key>NSAllowsArbitraryLoads</key>

<**true**/>

</dict>

3、如果在iOS9中使用了调起百度地图客户端功能，必须在"Info.plist"中进行如下配置，否则不能调起百度地图客户端。

<key>LSApplicationQueriesSchemes</key>

<array>

<string>baidumap</string>

</array>

4、管理地图的生命周期：自2.0.0起，BMKMapView新增viewWillAppear、viewWillDisappear方法来控制BMKMapView的生命周期，并且在一个时刻只能有一个BMKMapView接受回调消息，因此在使用BMKMapView的viewController中需要在viewWillAppear、viewWillDisappear方法中调用BMKMapView的对应的方法，并处理delegate，代码如下：

(**void**)viewWillAppear:(BOOL)animated

{

[\_mapView viewWillAppear];

\_mapView.delegate = self; *// 此处记得不用的时候需要置nil，否则影响内存的释放*

}

-(**void**)viewWillDisappear:(BOOL)animated

{

[\_mapView viewWillDisappear];

\_mapView.delegate = nil; *// 不用时，置nil*

}

5、自iOS SDK v2.5.0起，为了对iOS8的定位能力做兼容，做了相应的修改，开发者在使用过程中注意事项如下： 需要在info.plist里添加（以下二选一，两个都添加默认使用NSLocationWhenInUseUsageDescription）：

NSLocationWhenInUseUsageDescription ，允许在前台使用时获取GPS的描述

NSLocationAlwaysUsageDescription ，允许永久使用GPS的描述

6、在使用Xcode6进行SDK开发过程中，需要在info.plist中添加：Bundle display name ，且其值不能为空（Xcode6新建的项目没有此配置，若没有会造成manager start failed）

7、app在前后台切换时，需要使用下面的代码停止地图的渲染和openGL的绘制：

- (**void**)applicationWillResignActive:(UIApplication \*)application {

[BMKMapView willBackGround];*//当应用即将后台时调用，停止一切调用opengl相关的操作*

}

* (**void**)applicationDidBecomeActive:(UIApplication \*)application

{

[BMKMapView didForeGround];*//当应用恢复前台状态时调用，回复地图的渲染和opengl相关的操作*

}

二、配置开发包

第一步、根据需要导入 .framework包

将所需的BaiduMapAPI\_\*\*.framework拷贝到工程所在文件夹下。

在 TARGETS->Build Phases-> Link Binary With Libaries中点击“+”按钮，在弹出的窗口中点击“Add Other”按钮，选择BaiduMapAPI\_\*\*.framework添加到工程中。

第二步、引入所需的系统库

百度地图SDK中提供了定位功能和动画效果，v2.0.0版本开始使用OpenGL渲染，因此您需要在您的Xcode工程中引入CoreLocation.framework和QuartzCore.framework、OpenGLES.framework、SystemConfiguration.framework、CoreGraphics.framework、Security.framework、libsqlite3.0.tbd（xcode7以前为 libsqlite3.0.dylib）、CoreTelephony.framework 、libstdc++.6.0.9.tbd（xcode7以前为libstdc++.6.0.9.dylib）。

（注：红色标识的系统库为v2.9.0新增的系统库，使用v2.9.0及以上版本的地图SDK，务必增加导入这3个系统库。）

添加方式：在Xcode的Project -> Active Target ->Build Phases ->Link Binary With Libraries，添加这几个系统库即可。

第三步、环境配置

在TARGETS->Build Settings->Other Linker Flags 中添加-ObjC。

第四步、引入mapapi.bundle资源文件

如果使用了基础地图功能，需要添加该资源，否则地图不能正常显示mapapi.bundle中存储了定位、默认大头针标注View及路线关键点的资源图片，还存储了矢量地图绘制必需的资源文件。如果您不需要使用内置的图片显示功能，则可以删除bundle文件中的image文件夹。您也可以根据具体需求任意替换或删除该bundle中image文件夹的图片文件。

方法：选中工程名，在右键菜单中选择Add Files to “工程名”…，从BaiduMapAPI\_Map.framework||Resources文件中选择mapapi.bundle文件，并勾选“Copy items if needed”复选框，单击“Add”按钮，将资源文件添加到工程中。

第五步、引入头文件

在使用SDK的类 按需 引入下边的头文件：

#import <BaiduMapAPI\_Base/BMKBaseComponent.h>*//引入base相关所有的头文件*

#import <BaiduMapAPI\_Map/BMKMapComponent.h>*//引入地图功能所有的头文件* #import <BaiduMapAPI\_Search/BMKSearchComponent.h>*//引入检索功能所有的头文件*

#import <BaiduMapAPI\_Cloud/BMKCloudSearchComponent.h>*//引入云检索功能所有的头文件*

#import <BaiduMapAPI\_Location/BMKLocationComponent.h>*//引入定位功能所有的头文件*

#import <BaiduMapAPI\_Utils/BMKUtilsComponent.h>*//引入计算工具所有的头文件*

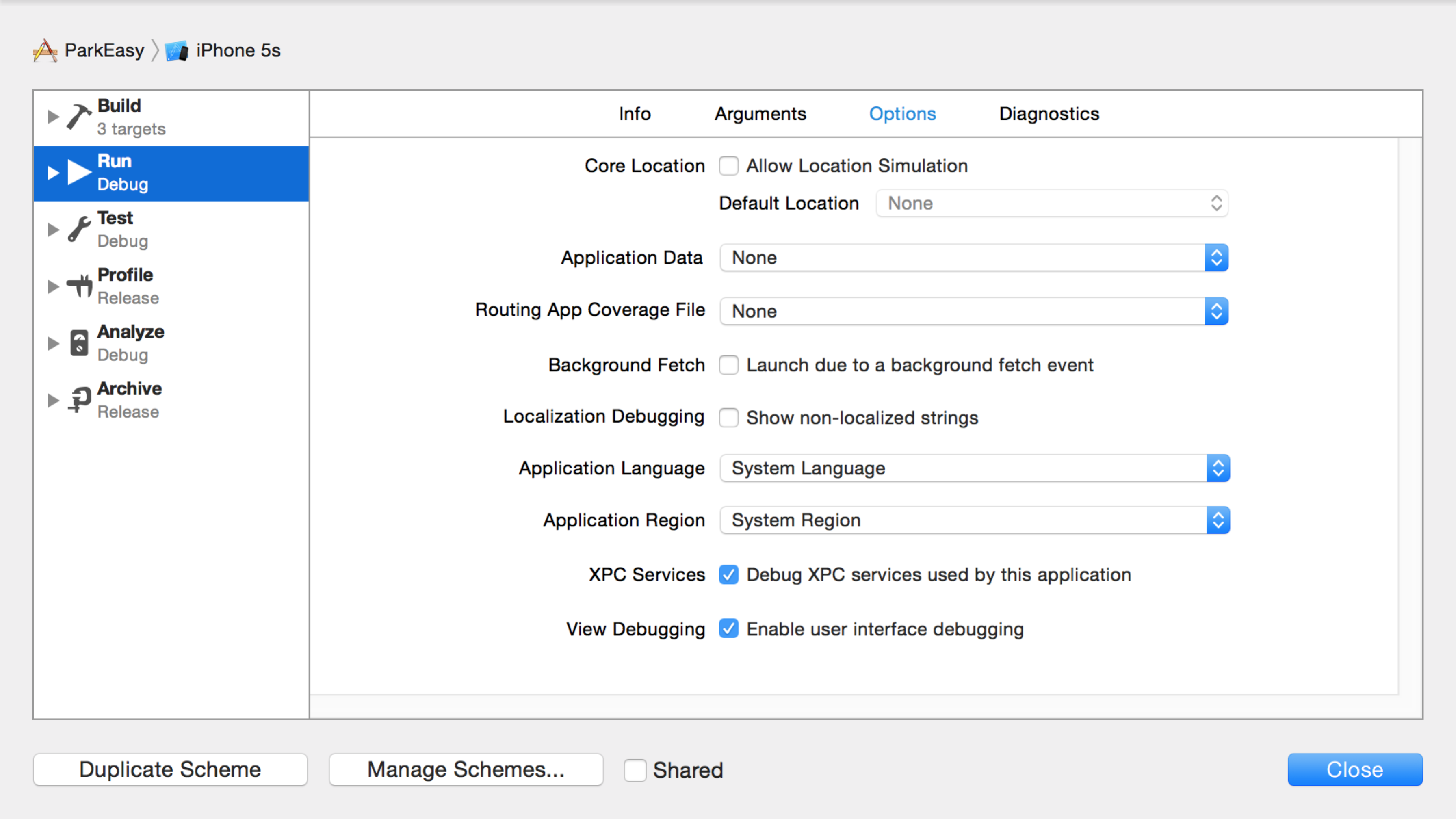
#import <BaiduMapAPI\_Radar/BMKRadarComponent.h>*//引入周边雷达功能所有的头文件*

#import < BaiduMapAPI\_Map/BMKMapView.h>*//只引入所需的单个头文件*

三、特别注意

模拟器不能定位的问题：

1)Edit Schemes 🡪 Run 🡪Option

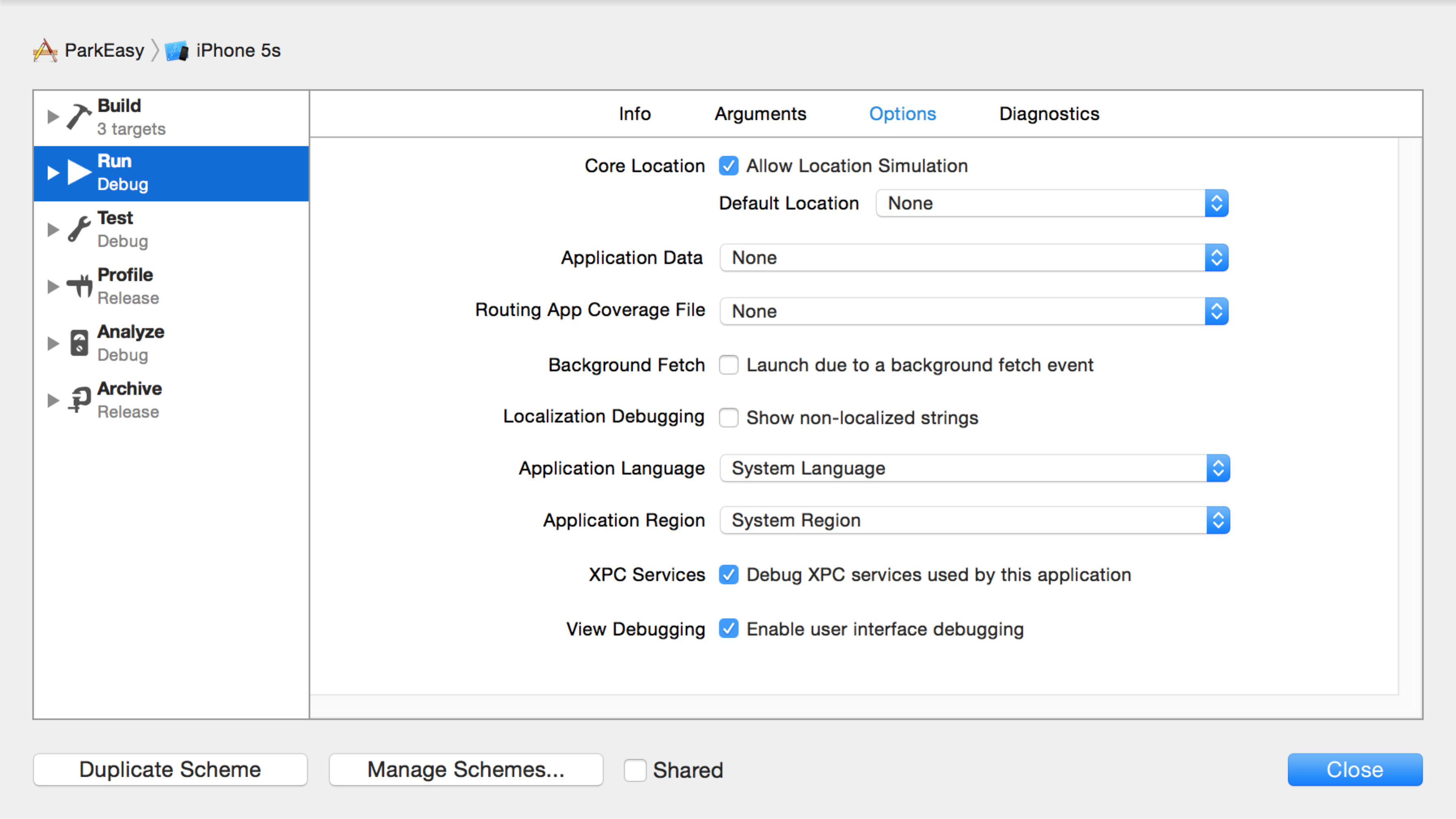


先把 Allow location Simulation前面的勾去掉

2）重启模拟器

Hardware 🡪 reboot

3) Edit Schemes 🡪 Run 🡪Option , 把 Allow location Simulation前面的勾选上



4）重启模拟器

Hardware 🡪 reboot

5）运行百度地图demo，点击定位，发现模拟器正常定位了。

**支付宝：**

一、注意事项

1、如果遇到运行后报错,类似于以下􏰁示信息:

Cannot find interface declaration for 'NSObject', superclass of 'Base64'

那么需要打开报错了的文件,增加头文件。

#import <Foundation/Foundation.h>

2、设置Head Search Paths 和 Library Search Paths。

targets->Build Setting ->Search Paths->Head Search Paths和 Library Search Paths

切记：Head Search Paths 和 Library Search Paths的路径一定要根据这个地址能找到对应的头文件。不然设置总会有这个错误（我就在这儿耗了很多时间），如：

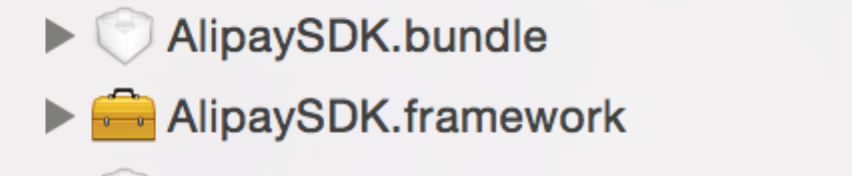
#include <openssl/rsa.h> 'openssl/rsa.h' file not found

二、配置开发包

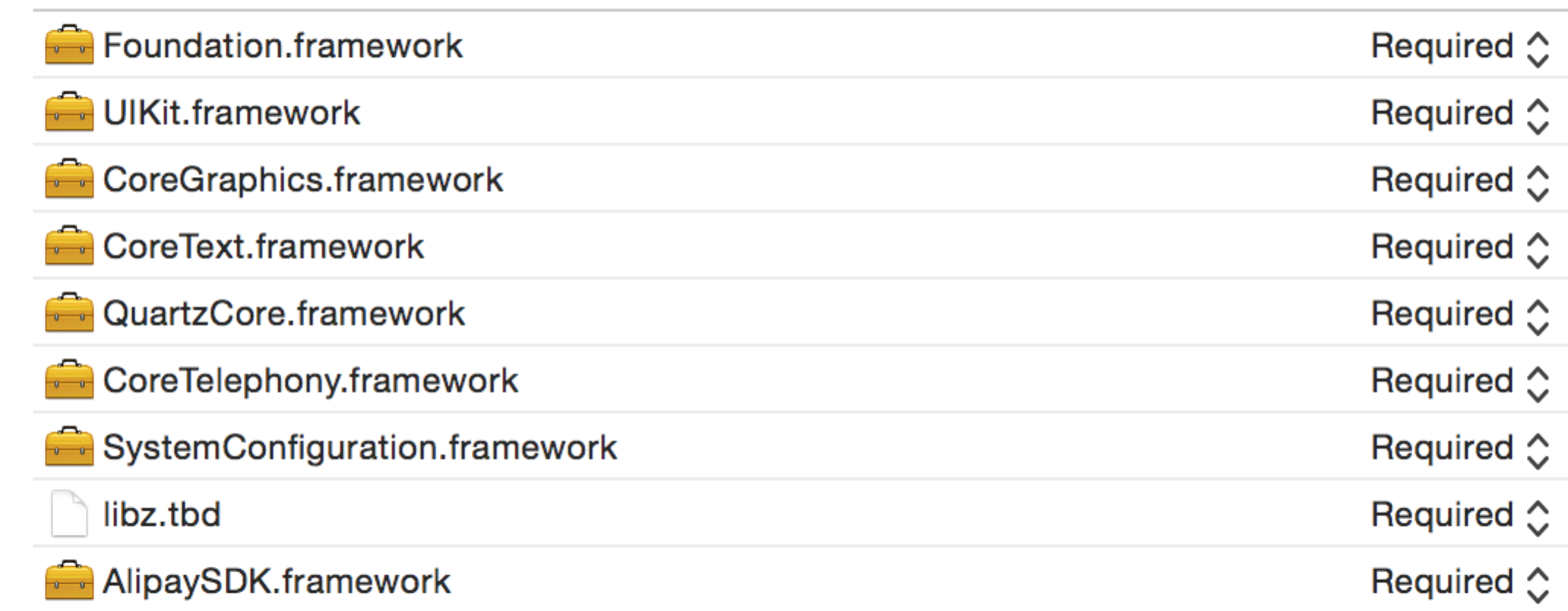
解压接口压缩文件(文件名是 WS\_MOBILE\_PAY\_SDK\_BASE.zip),找到 iOS 的压缩文件(文件名是支付宝钱包支付开发包标准版(iOS).zip)。

步骤**1**: 启动IDE(如Xcode),把iOS包中的压缩文件中以下文件拷贝到项目文件夹下,

并导入到项目工程中。



依赖添加完毕后,如下图所示结构。



步骤**2**: 在需要调用AlipaySDK的文件中,增加头文件引用。

#import <AlipaySDK/AlipaySDK.h>

步骤**3**: 为了适配 **iOS9.0** 中的 **App Transport Security(ATS)**对 **http** 的限制

请在"Info.plist"中进行如下配置，否则影响SDK的使用。

<key>NSAppTransportSecurity</key>

<dict>

<key>NSAllowsArbitraryLoads</key>

<**true**/>

</dict>

步骤**4**: 配置请求信息。

步骤**5**: 配置支付宝客户端返回url处理方法。

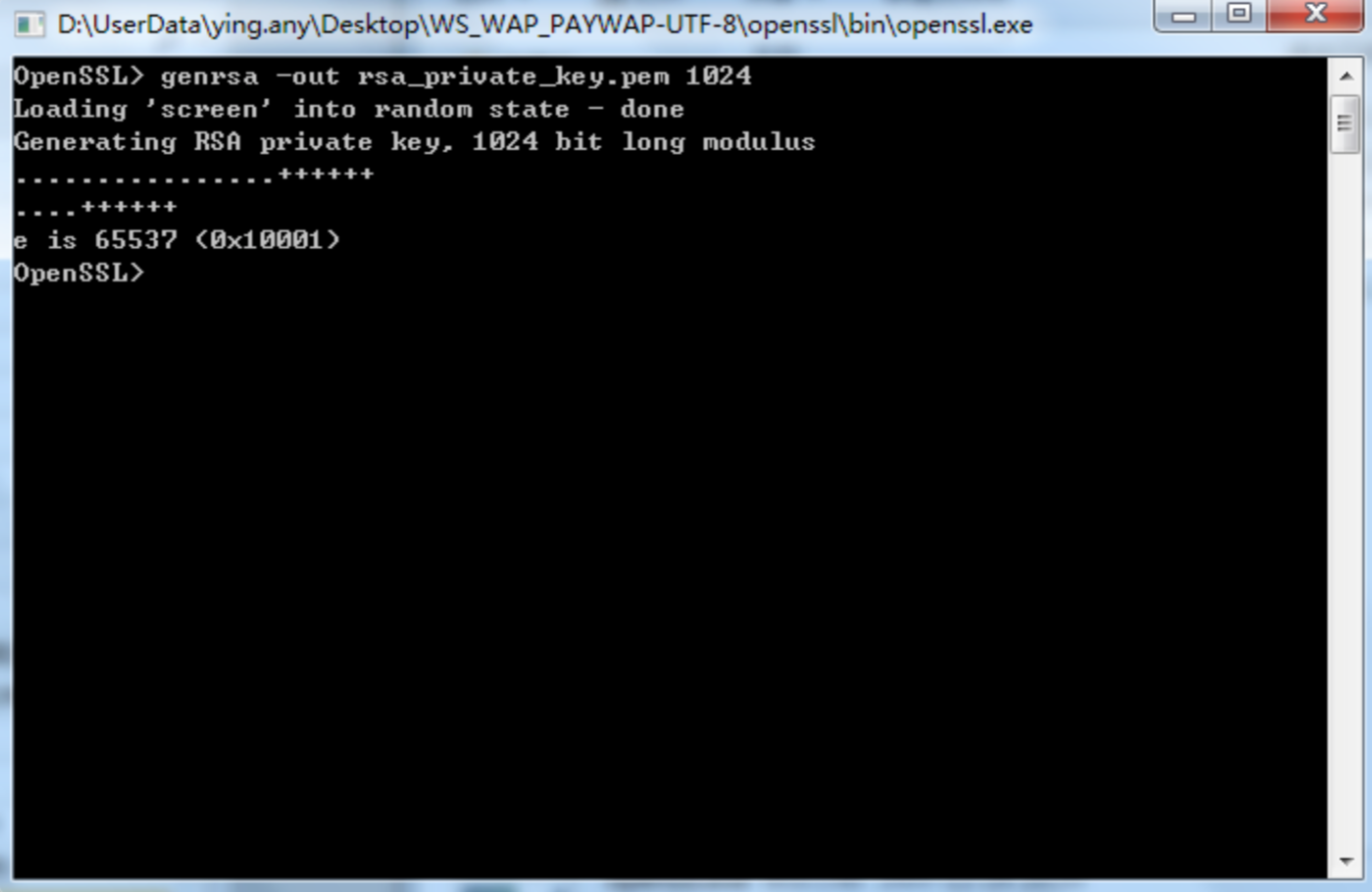
三、RSA密钥生成与使用

1. 打开openssl密钥生成软件

打开 openssl 文件夹下的 bin 文件夹,执行 openssl.exe 文件

2. 生成RSA私钥

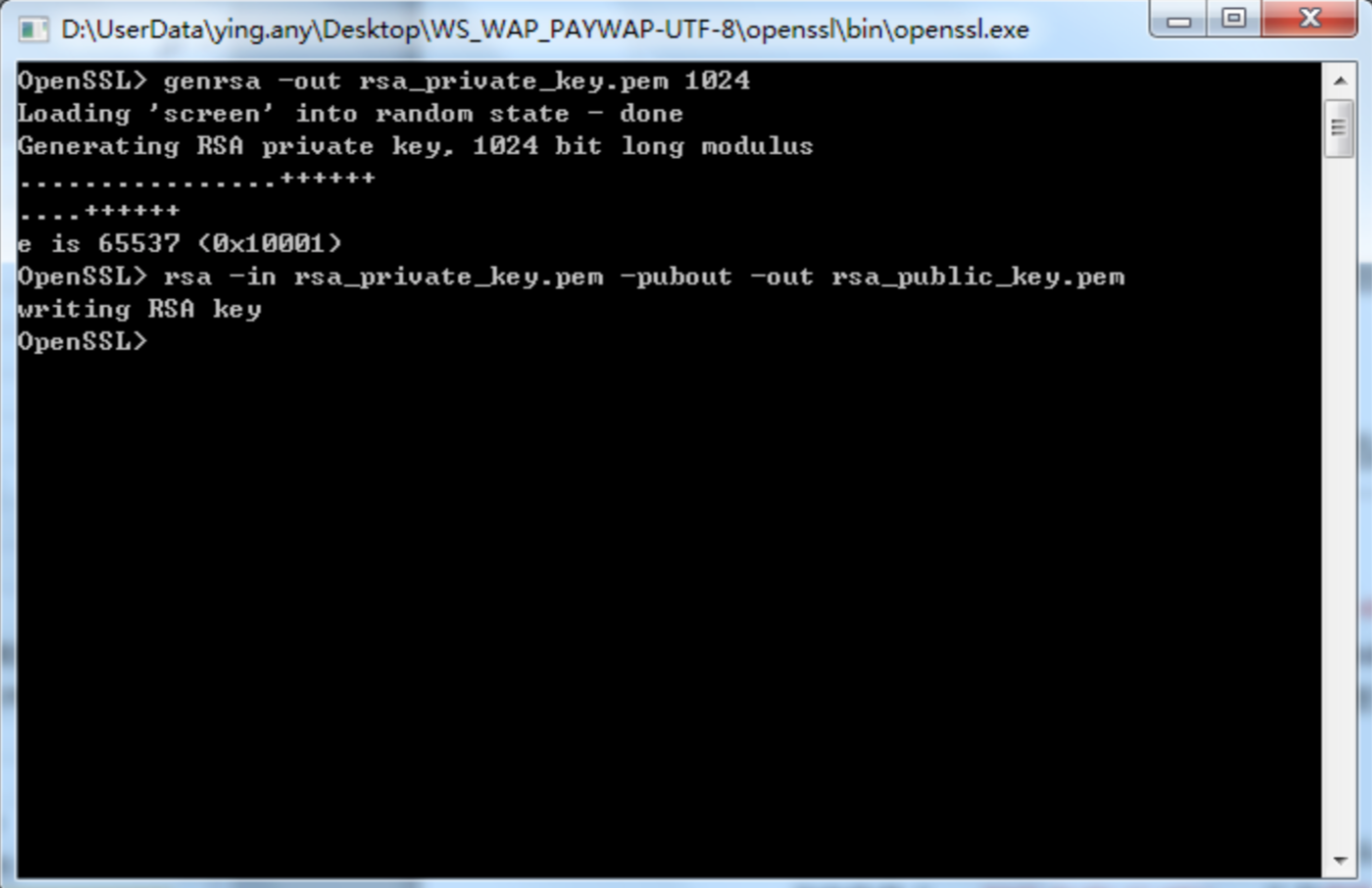
输入“*genrsa -out rsa\_private\_key.pem 1024*”命令,回车后,在当前 bin 文件目 录中会新增一个 rsa\_private\_key.pem 文件,其文件为原始的商户私钥(请妥善保 存该文件,PHP 开发语言中需要使用该文件) ,以下为命令正确执行截图:



3. 生成RSA公钥

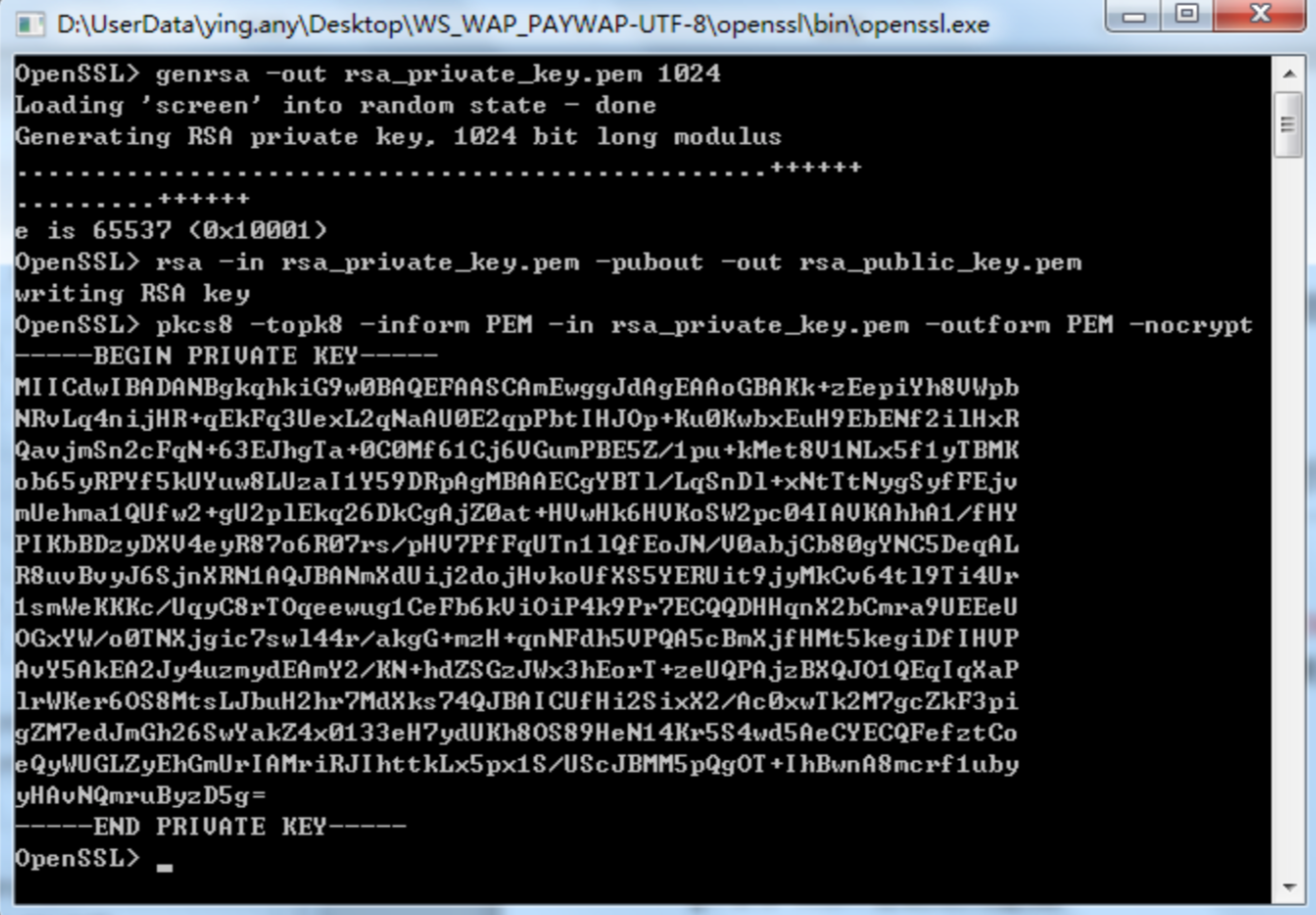
图7-8 生成 RSA 私钥

输入“*rsa -in rsa\_private\_key.pem -pubout -out rsa\_public\_key.pem*”命令回车 后,在当前 bin 文件目录中会新增一个 rsa\_public\_key.pem 文件,其文件为原始 的商户公钥(请妥善保存该文件,PHP 开发语言中需要使用该文件),以下为命 令正确执行截图:



4. 生成PKCS8 编码的私钥

输入命令“*pkcs8 -topk8 -inform PEM -in rsa\_private\_key.pem -outform PEM -nocrypt*”并回车,当前界面中会直接显示出生成结果:



四、特别注意

在上传公钥或设置privateKey时，一定要保证:

1、去掉字符串中的回车、换行、空格,变成只有一行文字;

2、去掉“-----BEGIN PUBLIC

KEY-----”、“-----END PUBLIC KEY-----”这两条文字;

踩坑：我在PC的记事本中看着是没有空格和换行的，但是我后来用NotePad++打开后，发现还存在空格和换行。删除之后，上传公钥成功。