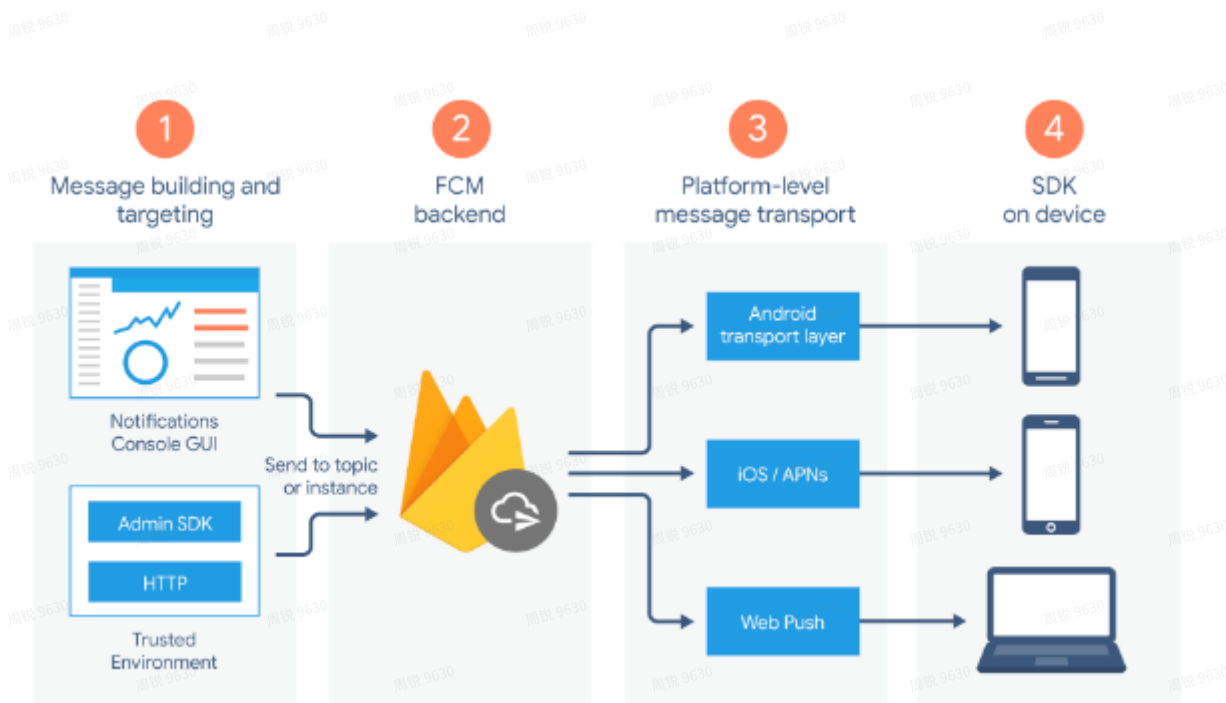


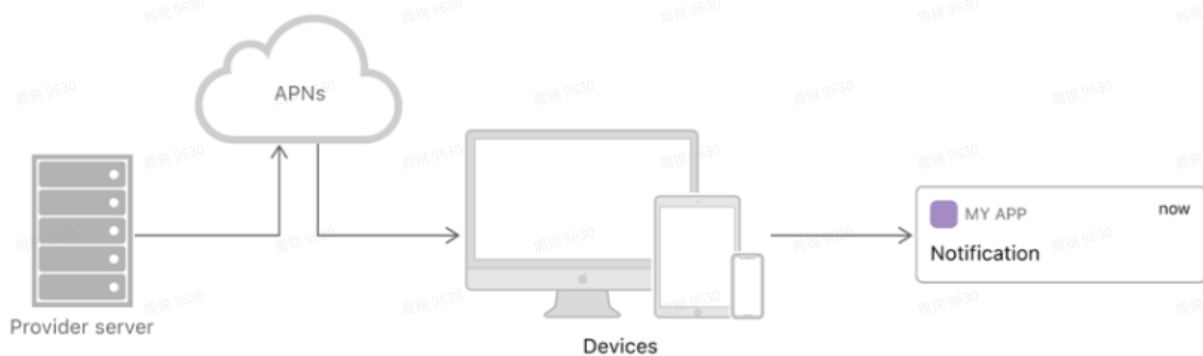
Firebase Messaging+APNs

一、Firebase Messaging原理简述

Firebase Messaging用于向手机发送push通知。当它针对IOS发送通知，相当于一个中间代理，将我们要发送的信息转发给苹果统一处理通知的APNs服务器。下面两张图片分别说明Firebase做了什么，APNs是什么。



Firebase作应用服务器-APNs间代理



APNs通知流程

在这个流程中，应用需要做什么：

- (1) 申请notification权限
- (2) 调用Firebase提供插件，将fcmToken（firebase识别当前用户该设备的标识）与通知配置传给应用服务器

(3) 调用Firebase提供插件，处理用户接收到通知并进入APP后的通知信息

应用服务器需要做什么：

- (1) 通过某些命令触发，构建通知消息，预备发送
- (2) 通过私钥，与Firebase中注册的应用（以bundle id标识）建立联系
- (3) 将消息通过HTTP/Firebase提供插件发送给Firebase服务器

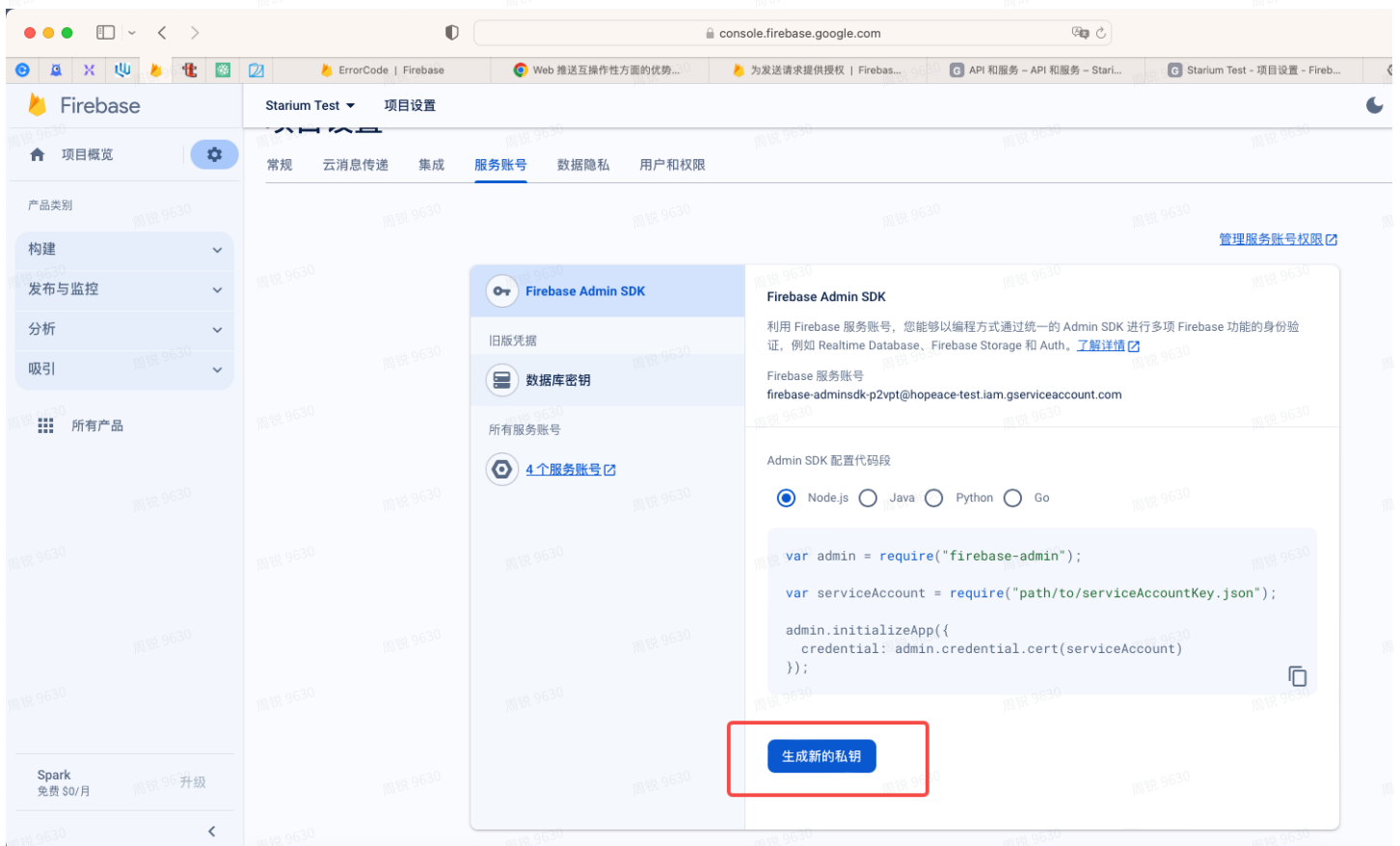
Firebase需要做什么：

- (1) 存储注册了fcmToken的用户到指定应用
- (2) 接受来自应用服务器的消息，如果带有fcmToken，需要与存储里的token匹配
- (3) 将消息发给APNs,通过开发者p8私钥与APNs进行验证通信，与之对应的，应用需要是基本信息相同（团队、bundle等）的p12证书。

二、通过firebase_admin插件模拟应用服务器向firebase发送通知消息

以python为例，为了模拟测试需要做如下准备：

- (1) 引入firebase_admin插件
- (2) 相应App的SDK服务器私钥（一个json文件）



(3) 从客户端获取fcmToken

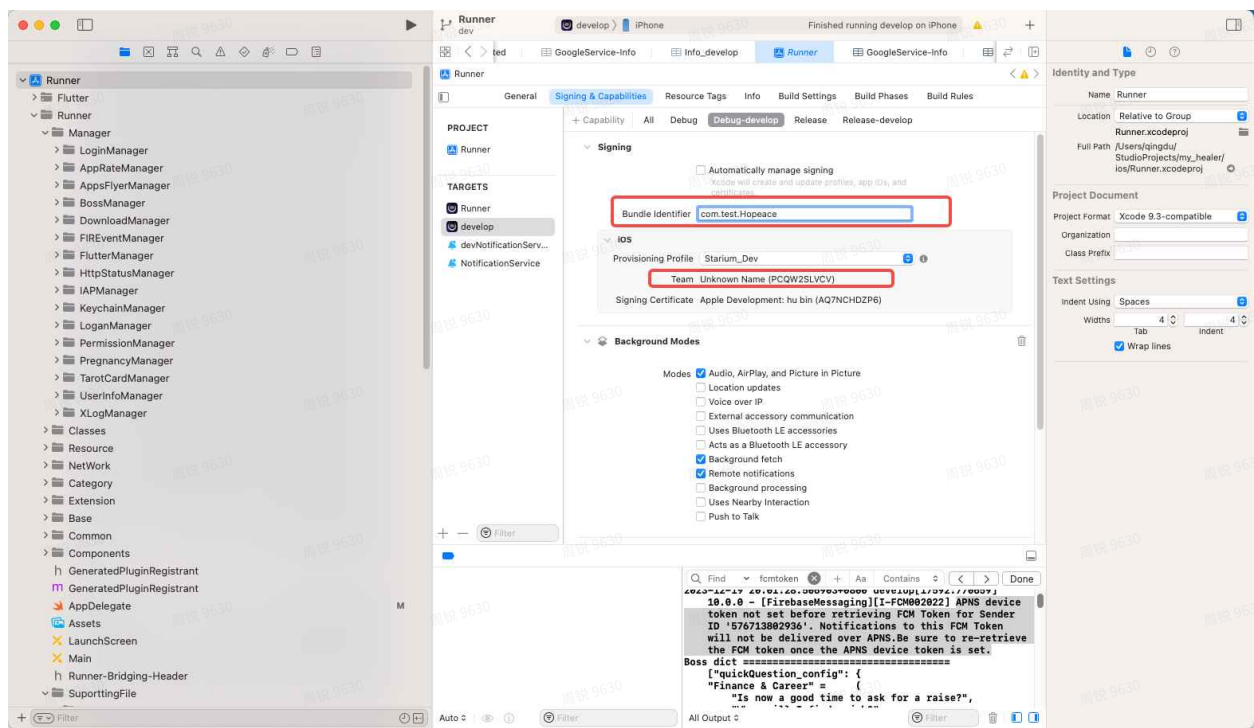
测试流程可以分为以下几步：

- (1) 引入官方测试代码[Firebase Messaging 测试代码](#)
- (2) 在终端执行"export GOOGLE_APPLICATION_CREDENTIALS="path/to/service_account.json"隐式请求firebase授权或者参考[Firabase授权](#)
- (3) 将测试代码中的registration_token替换为你得到的fcmToken
- (4) 运行代码，如果有报错可参考[FireBase Messaging 常见错误码](#)

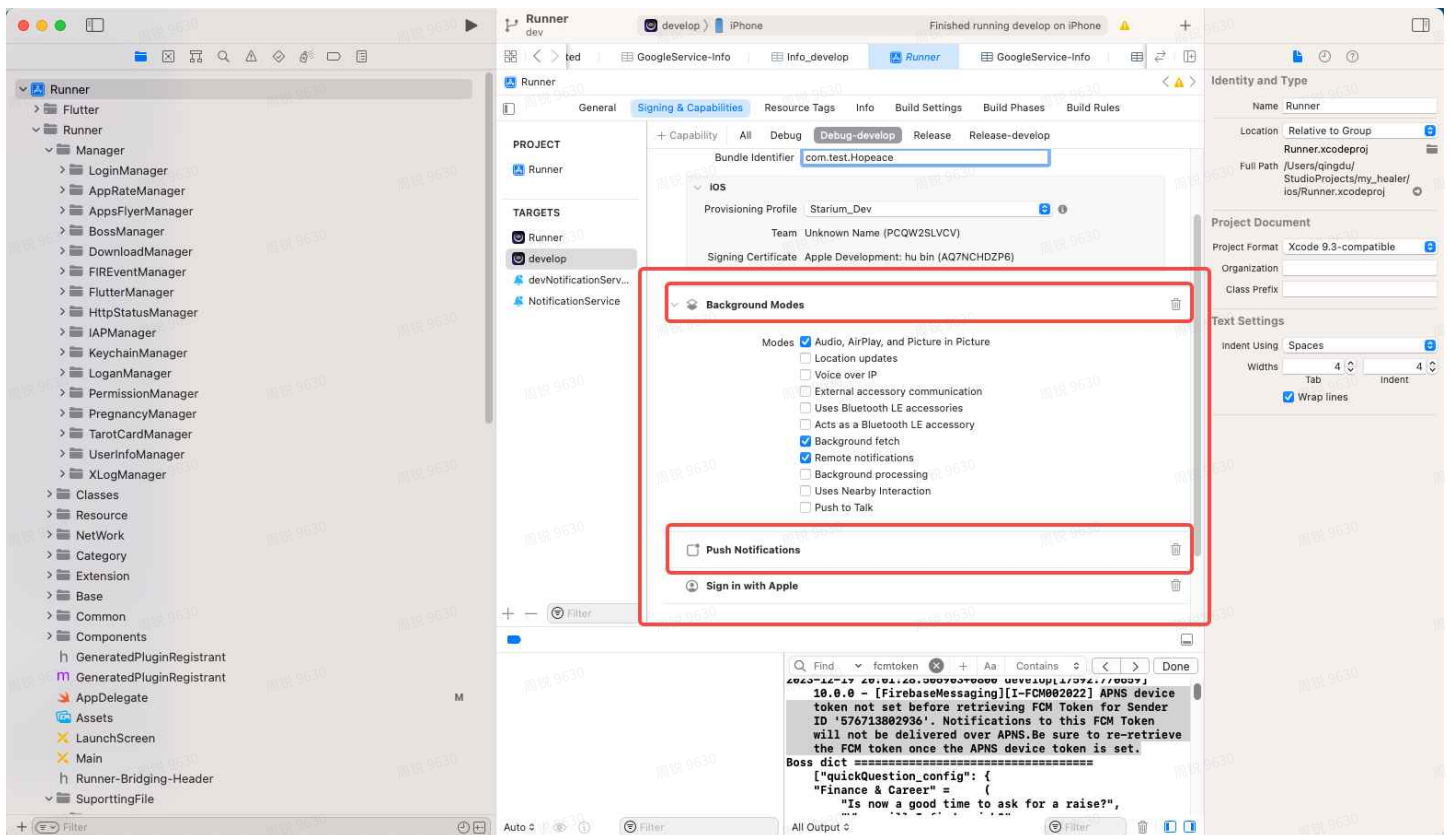
三、注意事项

- (1) firebase中的常规信息，需要与Xcode中的信息相匹配，团队等信息更新后，需要重新下载GoogleService-Info.plist文件。

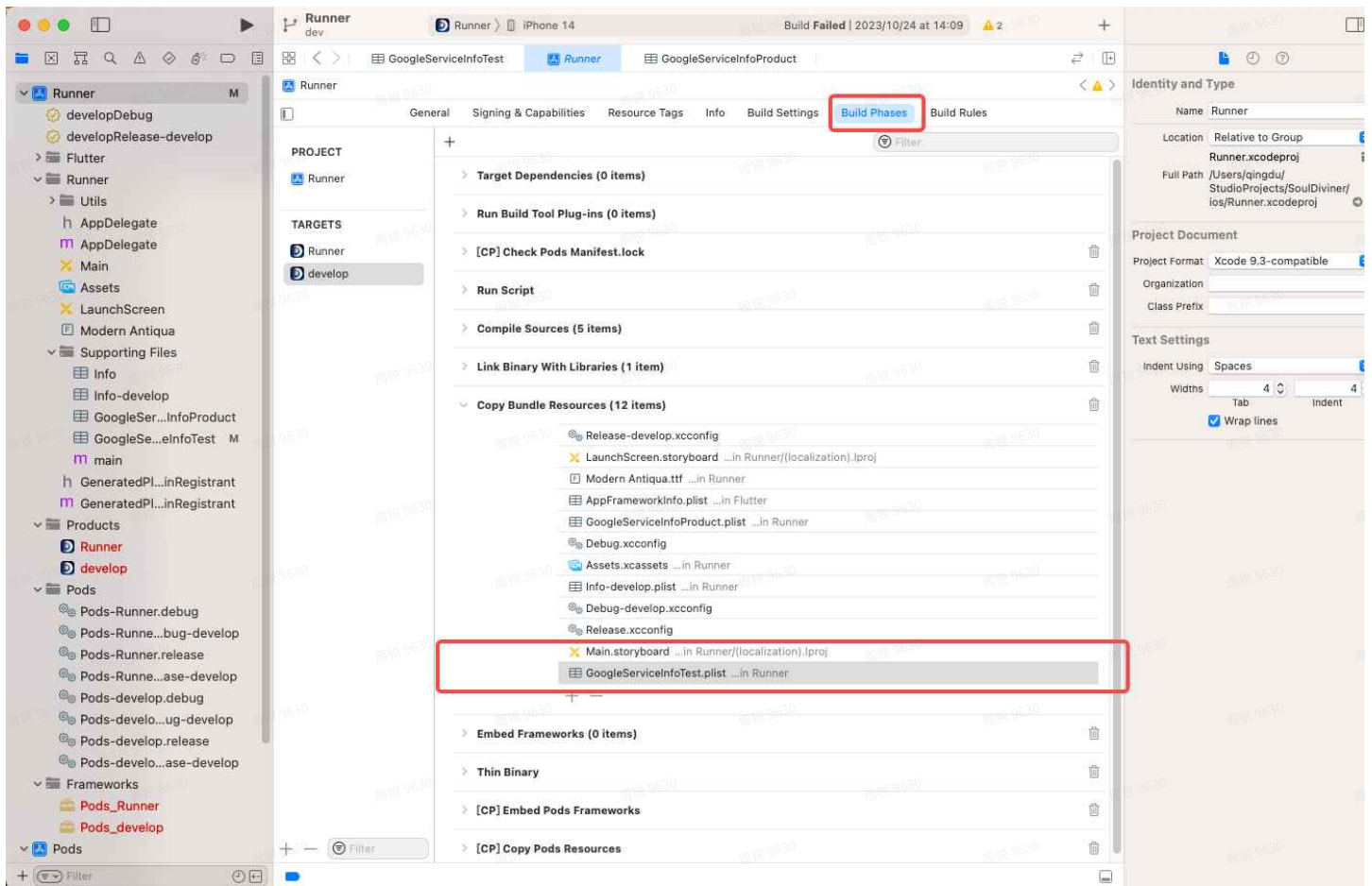




(2) 在客户端方面，为了给app引入消息通知功能,需要通过apple开发者账号创建相应的应用证书，并向其中添加notification service。然后开启BackGround Modes 与Push Notifications。



(3) 谷歌服务info文件在Xcode中的关联位置



(4) 应用角标设置(AppDelegete内)

```

- (void)applicationDidBecomeActive: (UIApplication *)application {
    [[UIApplication sharedApplication] setApplicationIconBadgeNumber:0];
    NSLog(@"applicationDidBecomeActive");
    [Tools requestAccessATTTracking:^(bool granted) {}];
}

- (void)applicationWillEnterForeground:(UIApplication *)application {
    [[UIApplication sharedApplication] setApplicationIconBadgeNumber:0];
}

```

objc

```

override func applicationDidBecomeActive(_ application: UIApplication) {
    UIApplication.shared.applicationIconBadgeNumber = 0
    AppsFlyerManager.shared.requestTrackingAuthorization()
}

override func applicationWillEnterForeground(_ application: UIApplication) {
    UIApplication.shared.applicationIconBadgeNumber = 0
}

```

swift