信鸽推送接入小结

特点

- 信鸽推送整合iOS8+的推送兼容，推送处理比较简单
- 不需要自己注册提交deviceToken
- 只能绑定一个账号，标签可以多个。每个应用下最多有1万个标签，每个设备最多绑定1百个标签。
- 批量设备推送单次最多推送给1000个设备token，批量帐号推送单次最多推送给1000个帐号。

接入步骤

- 导入sdk，引用头文件，具体看官方文档
- 重点步骤，看如下代码：

```
// 注: App ID 对应应用 Access ID, App Key 对应应用 Access Key.
@interface AppDelegate () <XGPushDelegate>
@end

-(BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions {
    [[XGPush defaultManager] startXGWithAppID:<#your AppID#> appKey:<#your appKey#>  delegate:<#your delegate#>];
    [[XGPush defaultManager] setXgApplicationBadgeNumber:0];
    [[XGPush defaultManager] reportXGNotificationInfo:launchOptions];
    return YES;
}
```

- 处理上报

```
// ios 8-9 在前后台接到推送都会走
- (void)application:(UIApplication *)application didReceiveRemoteNotification:(NSDictionary *)userInfo
fetchCompletionHandler:(void (^)(UIBackgroundFetchResult))completionHandler {
    [[XGPush defaultManager] reportXGNotificationInfo:userInfo];
    completionHandler(UIBackgroundFetchResultNewData);
}
```

- 处理点击事件

```
#if __IPHONE_OS_VERSION_MAX_ALLOWED >= __IPHONE_10_0
- (void)xgPushUserNotificationCenter:(UNUserNotificationCenter *)center didReceiveNotificationResponse:(UNNotificationResponse *)resp
withCompletionHandler:(void (^)(void))completionHandler {
    // 上报点击事件
    [[XGPush defaultManager] reportXGNotificationResponse:response];
    completionHandler();

    // ios 10+ 前后台点击都走这里
    // 处理自己的跳转
}

// App 在前台弹通知需要调用这个接口，以前iOS8，9是不会弹通知的，在前台不弹出在这里控制
- (void)xgPushUserNotificationCenter:(UNUserNotificationCenter *)center willPresentNotification:(UNNotification *)notification
withCompletionHandler:(void (^)(UNNotificationPresentationOptions))completionHandler {
    [[XGPush defaultManager] reportXGNotificationInfo:notification.request.content.userInfo];
    completionHandler(UNNotificationPresentationOptionBadge | UNNotificationPresentationOptionSound | UNNotificationPresentationOptio
}
#endif

- (void)xgPushDidReceiveRemoteNotification:(id)notification withCompletionHandler:(void (^)(NSUInteger))completionHandler {
    if ([notification isKindOfClass:[NSDictionary class]]) {
        // 上报数据
        [[XGPush defaultManager] reportXGNotificationInfo:(NSDictionary *)notification];
        completionHandler(UIBackgroundFetchResultNewData);

        // ios 9 在前后台都会走此方法
        if(!([UIApplication sharedApplication].applicationState == UIApplicationStateActive)){
            // 处理自己的跳转
        }
    } else if ([notification isKindOfClass:[UNNotification class]]) {
        // 上报数据
        [[XGPush defaultManager] reportXGNotificationInfo:((UNNotification *)notification).request.content.userInfo];
        completionHandler(UNNotificationPresentationOptionBadge | UNNotificationPresentationOptionSound | UNNotificationPresentationO
        // 似乎没啥用?
        //ios 10,ios 11 前台会走这里
        if(!([UIApplication sharedApplication].applicationState == UIApplicationStateActive)){
            // 处理自己的跳转
        }
    }
}
```

- 其他api自己看文档了解

生成签名

```
openssl pkcs12 -in CertificateName.p12 -out CertificateName.pem -nodes
```