

# iOS SDK Objective-c 安装文档

## 一、登陆 OneAPM 后台

登陆 **Mi SaaS** 平台账号 ([点击登录](#)) 或企业版后台 (企业版后台地址请咨询**OneAPM**工作人员)。

## 二、创建应用项目

1. 在应用程序列表页面，点击“创建”按钮。

2. 输入应用名称，点击保存。

**注意：**包名不同的应用不可使用同一 **Token**。

应用名称可自定义，中文、英文、数字均可，建议不使用重复的 **APP** 名称，可在设置模块对已有的 **APP** 名称进行修改。

3. 随后会获取应用的唯一标识 **Token** 值，用于配置 **sdk**。

### 三、集成 SDK

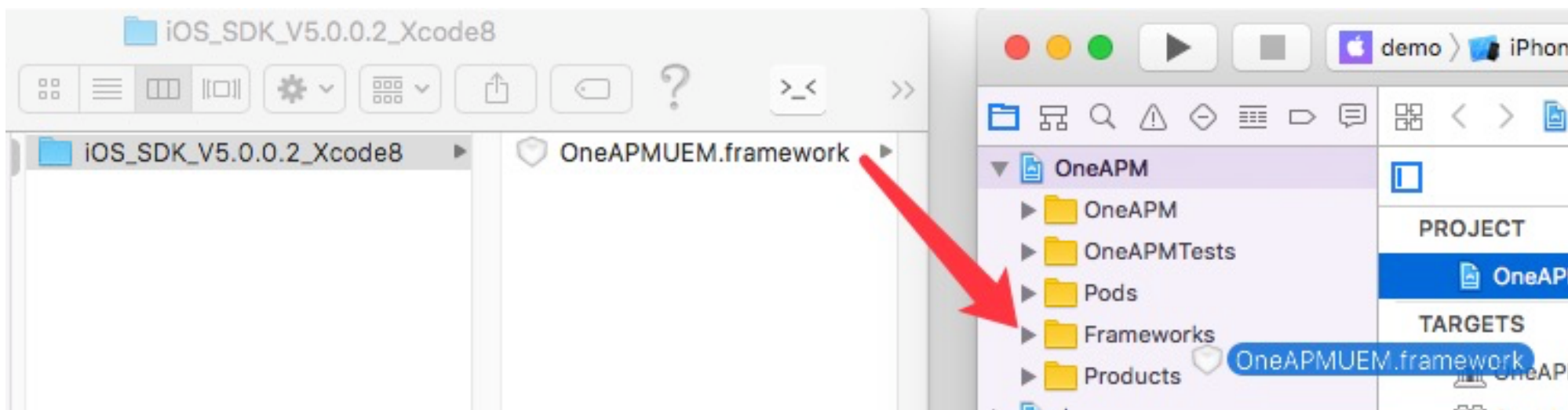
#### 1. 准备 iOS SDK。

从官网或OneAPM工作人员取得SDK。

#### 2. 添加 OneAPMUEM.Framework 至 Xcode 项目中

解压 SDK，并将「OneAPMUEM.framework」文件夹从 Finder 中拖拽至 Xcode 项目中（悬停至导航窗口的项目中）。出现提示窗口时，选择「Copy items into destination...」和「Create folder references...」。

如下图所示：

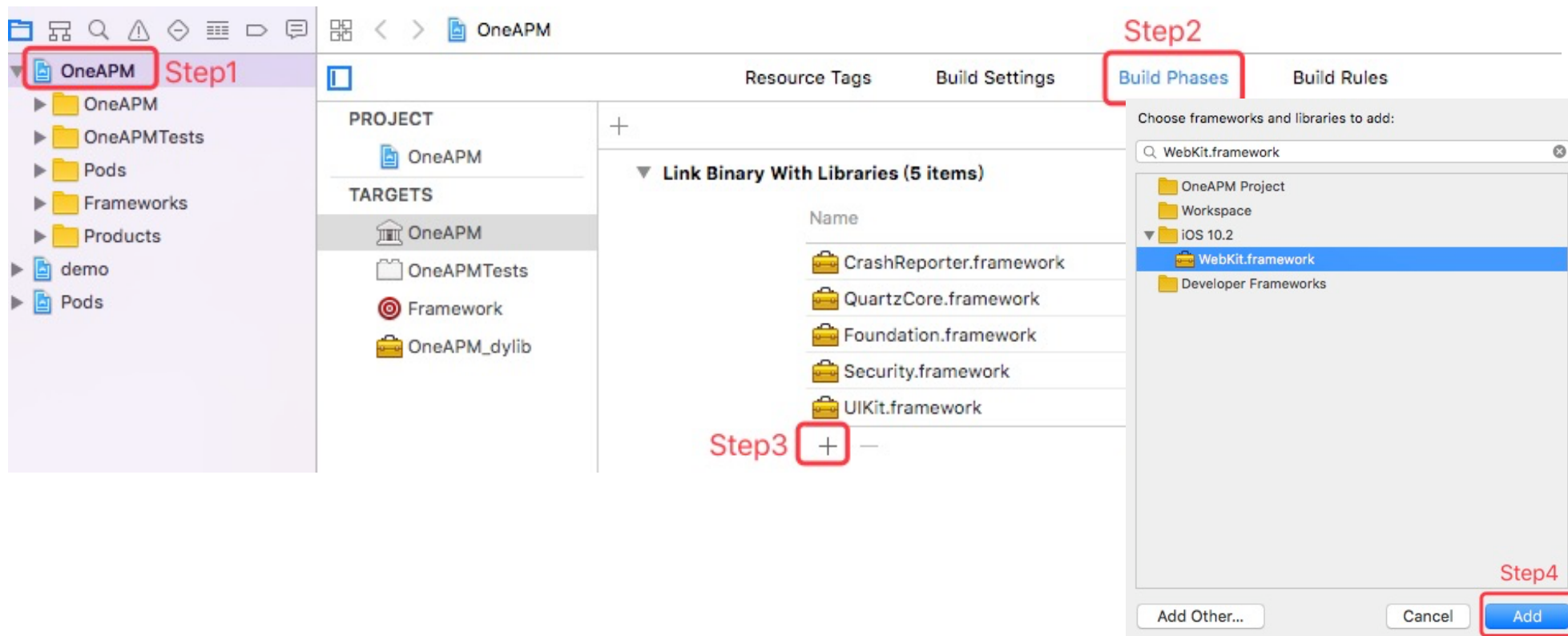


### 3. 在 Linker Settings 中添加以下 5 个 Libraries

在项目导航窗口内点击你的 **Project**，并选中你的**App**，然后选择「**Build Phases**」选项卡。

打开「**Link Binary with Libraries**」列表，点击添加：

- [SystemConfiguration.framework](#)
- [CoreTelephony.framework](#)
- [libz.tbd\ libz.dylib](#) (Xcode6及以前版本)
- [libstdc++.tbd\ libstdc++.dylib](#) (Xcode6及以前版本)
- [WebKit.framework](#)



## 4. 引入SDK

- (1) 在项目文件 [app\_name]-Prefix.pch（通常在文件夹「Support Files」中）或者 main.m 文件中，引入 OneAPM 头文件：

```
#import <OneAPMUEM/OneAPM.h>
```

- (2) 在 main.m 文件的 main 函数中添加如下代码：

```
int main(int argc, char * argv[]) {
```

```
    @autoreleasepool {
```

```
        1. OneAPM log 开关设置（可选）
```

```
        // 可以通过这句话打印OneAPM的Log，当集成以后出现类似“OneAPM SDK init succeed.”的Log，  
        说明集成成功。
```

```
        [OneAPM setPrintLog:YES];
```

```
        2. OneAPM 数据压缩 开关设置（可选）
```

```
        // 可以通过这句话来选择数据是否经过压缩上报，SDK 默认是压缩后上报数据，可以通过下面的代码  
        设置成不压缩直接上报数据。
```

```
        [OneAPM disableZlib];
```

### 3. 自定义配置用户信息（可选）

// 为了更好的展示用户访问信息，开发者可以在 SDK 启动代码前通过 **setUserName** 来自定义设置用户的信息，效果图请参考附录一：

**[OneAPM setUserName:@"phonenum"];**

注意：

1. **setUserName** 信息用户可自定义，常见配置邮箱、电话等
2. 如未设置该项，用户信息默认展示 **DeviceID**

### 4. 设置 MI dc 服务器

// 为了更方便的设置 MI dc服务器、端口号以及http / https协议，开发者可以在 SDK 启动代码前通过 **setHost**、**port**、**withSecurity**来自定义设置。

// **setHost** 可设置为域名或 IP 地址，二选一即可。

// **port**：设置为服务器端口号

// **withSecurity ==YES**：与服务器的通信使用 **https** 协议(根据苹果的审核要求，推荐使用 **https** 协议);

**withSecurity ==NO**：与服务器的通信使用 **http** 协议;

// 4.1 不设置端口

**[OneAPM setHost:@"mi dc domain or ip address" withSecurity:YES];**

// 4.2 设置端口

**[OneAPM setHost:@"mi dc domain or ip address" port:端口号 withSecurity:YES];**

## 5. token 嵌入

// 5.1 若使用线上服务器，token 嵌入方式如下：

```
[OneAPM startWithApplicationToken:@" <use app token created at step 2>"];
```

// 5.2 若使用企业级服务器，token 嵌入方式如下：

· 企业级应用如果需要送审的话，企业级部署时必须提供对 https 的支持，嵌入方式如下：

```
[OneAPM setHost:@"mi dc domain or ip address" withSecurity:YES];
```

```
[OneAPM startWithApplicationToken:@" <use app token created at step 2>"];
```

· 企业级应用如果不需要送审的话，企业级部署时可以不必要提供对 https 的支持，嵌入方式如下：

```
[OneAPM setHost:@"mi dc domain or ip address" withSecurity:NO];
```

```
[OneAPM startWithApplicationToken:@" <use app token created at step 2>"];
```

```
return UIApplicationMain(argc, argv, nil, NSStringFromClass([AppDelegate class]));
```

```
}
```

```
}
```

综合以上步骤，main.m 文件的 main 函数通常配置如下：

```
int main(int argc, char * argv[]) {
```

```
    @autoreleasepool {
```

```
        [OneAPM setPrintLog:YES];
```

```
        [OneAPM disableZlib];
```

```
        [OneAPM setUsername:@"phonenum"];
```

```
        [OneAPM setHost:@"mi dc domain or ip address" port:端口号 withSecurity:YES];
```

```
        [OneAPM startWithApplicationToken:@" <use app token created at step 2>"];
```

```
        return UIApplicationMain(argc, argv, nil, NSStringFromClass([AppDelegate class]));
```

```
    }
```

```
}
```

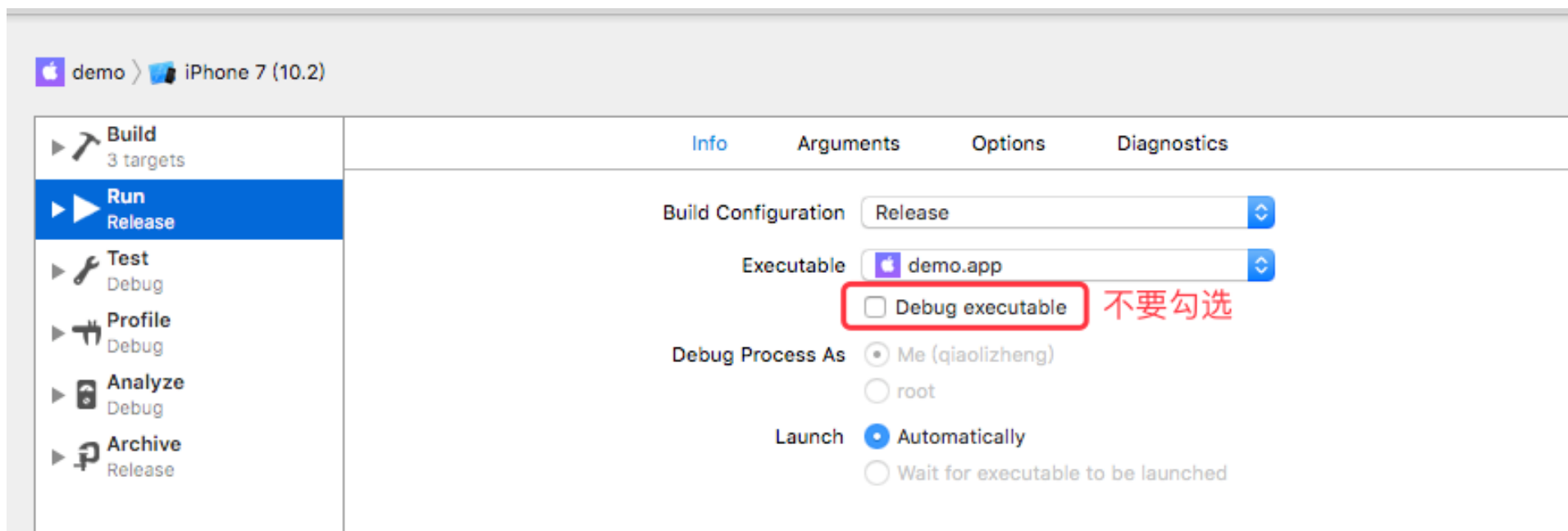
## 5. 运行应用程序并使用

**Clean Project**，并重新在模拟器或设备中启动应用程序，开始应用性能管理。

请静候1分钟，等待应用程序向 **OneAPM** 发送应用程序性能数据，即可开始使用 **OneAPM** 应用性能管理功能。

注意：

开发时若要使用“崩溃统计”功能，请去掉**Debug executable**选项并在真机上运行，或者断开真机与Mac的连接。



## 6. 符号表 (dSYM) 文件配置

为了正常解析崩溃信息，请按照以下步骤配置 **dSYM** 文件：

- (1) 手动上传：点击查看 [上传 dSYM 文件手动上传步骤](#)。
- (2) 自动上传：点击查看 [上传 dSYM 文件自动上传步骤](#)。