

Injection for Xcode 原理与使用

董桢远

donganyuan@baidu.com

2016.06.27

提纲

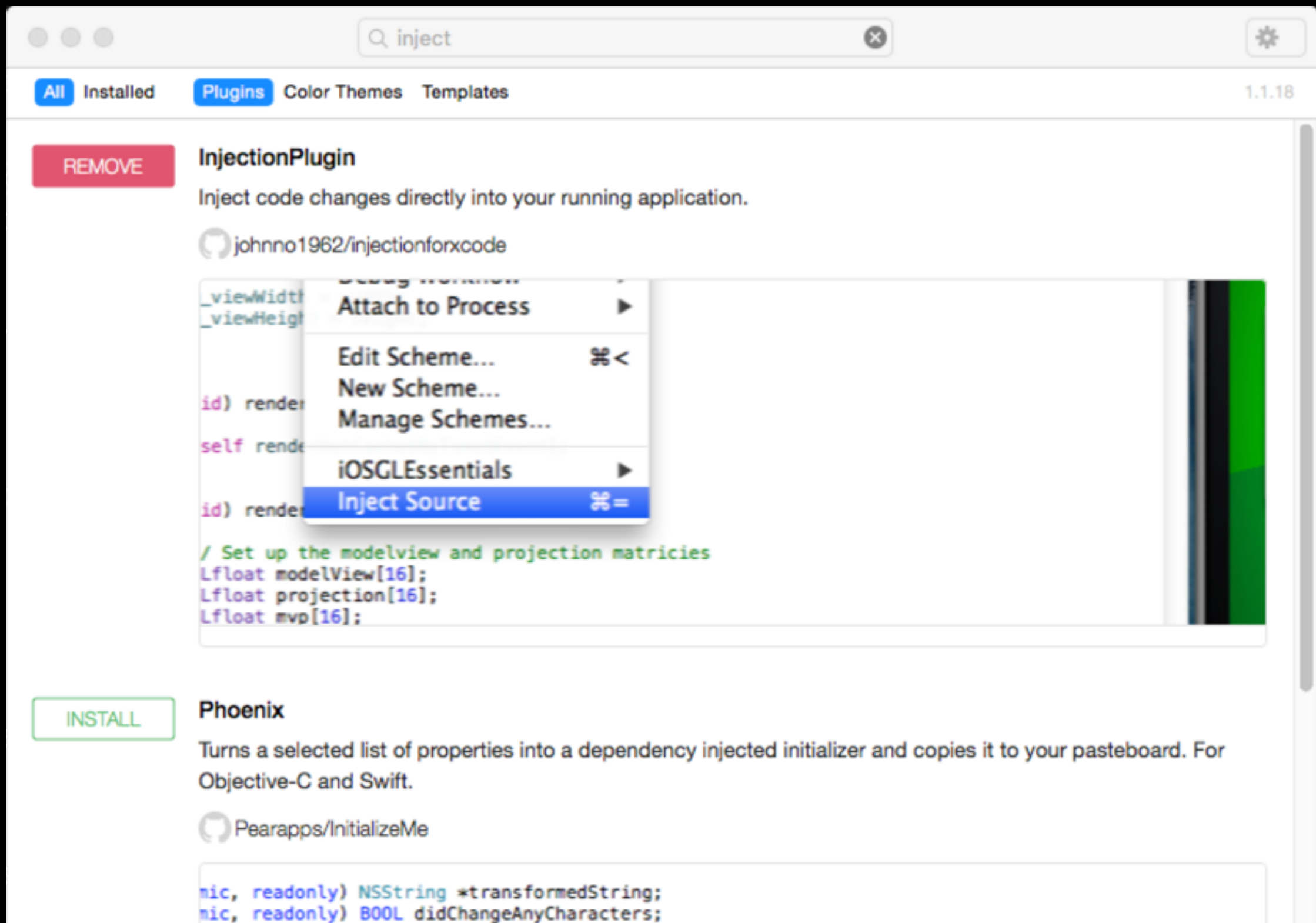
- 插件安装与删除
- 原理简述
- 使用
- 局限

提纲

- 插件安装与删除
- 原理简述
- 使用
- 局限

插件安装与删除

Alcatraz



插件安装与删除

手动安装

git clone git@github.com:johnno1962/
injectionforxcode.git 到本地，使用Xcode编译安
装后，重启Xcode加载插件即可。

插件安装与删除

删除插件

```
rm -rf ~/Library/Application\ Support/Developer/  
Shared/Xcode/Plug-ins/InjectionPlugin.xcplugin
```

提纲

- 插件安装与删除
- 原理简述
- 使用
- 局限

原理简述

0. 通过分析device log来分析哪些文件需要编译，放入新的bundle中。

1. 通过Application的动态 loader将新生成的bundle注入到应用中。

2. 将新生成的bundle替换旧bundle。

3.通过runtime 将bundle中的新代码swizzle老代码，这样再次生成新对象时，达到执行新代码的效果。

关键代码解析

+load

建立当前Xcode主机与调试器或真机的socket通信连接

+bundleLoader

Xcode与真机或调试器之间的命令对应

- loadBundle
- WiFi keepalive
- open file/directory to write/create
- open file/directory to read/list
- ◦ ◦ ◦

+loadBundle

- 加载bundle, dlopen, dlsym
- registerSelectorsInLibrary: containing:

提纲

- 插件安装与删除
- 原理简述
- 使用
- 局限

使用

注意点

- 添加新文件后，需要重新编译

Demo

使用

注意点

- 真机调试遇到过的问题

如果为纯swift工程，需要添加main.m文件，然后在文件中添加如下代码（OC工程中直接加入如下代码），详见demo工程TestInjectionXcode

```
#import <Foundation/Foundation.h>

// From here to end of file added by Injection Plugin //

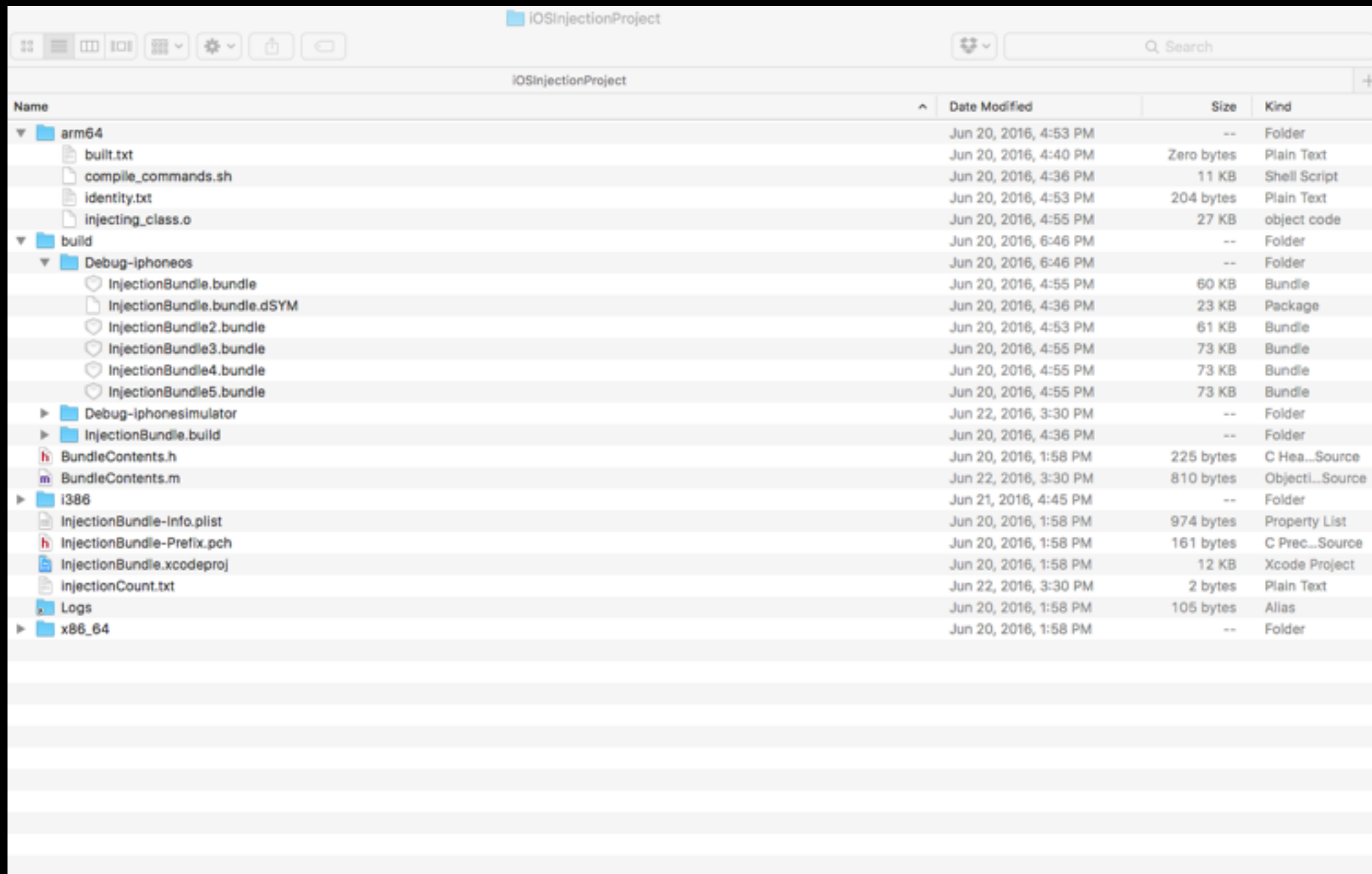
#ifdef DEBUG
static char _inMainFilePath[] = __FILE__;
//其中172.24.80.161为当前调试电脑的局域网IP
static const char *_inIPAddresses[] = {"172.24.80.161", "127.0.0.1", 0};

#define INJECTION_ENABLED
#import "/tmp/injectionforxcode/BundleInjection.h"
#endif
```

使用

注意点

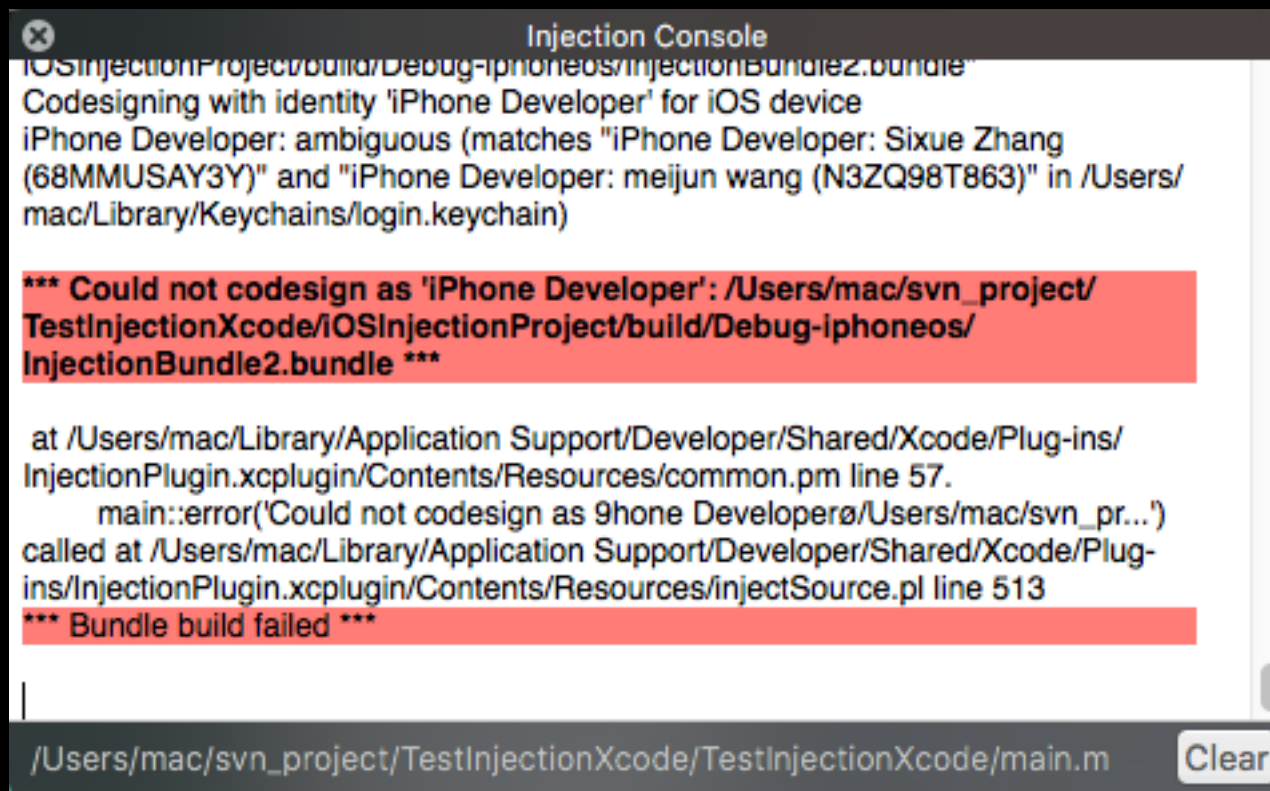
- 会在工程项目中生成iOSInjectionProject, 建议在svn或git ignore中在代码管理工具中忽略此文件



使用

注意点

- 真机调试如果出现

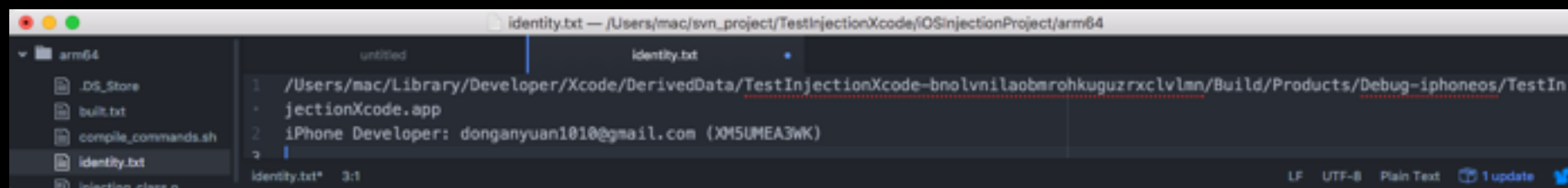


The screenshot shows the 'Injection Console' window with the following text:

```
IOSInjectionProject/build/Debug-iphoneos/InjectionBundle2.bundle  
Codesigning with identity 'iPhone Developer' for iOS device  
iPhone Developer: ambiguous (matches "iPhone Developer: Sixue Zhang  
(68MMUSAY3Y)" and "iPhone Developer: meijun wang (N3ZQ98T863)" in /Users/  
mac/Library/Keychains/login.keychain)  
  
*** Could not codesign as 'iPhone Developer': /Users/mac/svn_project/  
TestInjectionXcode/IOSInjectionProject/build/Debug-iphoneos/  
InjectionBundle2.bundle ***  
  
at /Users/mac/Library/Application Support/Developer/Shared/Xcode/Plug-ins/  
InjectionPlugin.xcplugin/Contents/Resources/common.pm line 57.  
main::error('Could not codesign as iPhone Developer: /Users/mac/svn_pr...')  
called at /Users/mac/Library/Application Support/Developer/Shared/Xcode/Plug-  
ins/InjectionPlugin.xcplugin/Contents/Resources/injectSource.pl line 513  
*** Bundle build failed ***
```

At the bottom, the file path `/Users/mac/svn_project/TestInjectionXcode/TestInjectionXcode/main.m` is shown next to a 'Clear' button.

将developer按照build setting中填全



The screenshot shows a text editor window with the file `identity.txt` open. The content of the file is:

```
1 /Users/mac/Library/Developer/Xcode/DerivedData/TestInjectionXcode-bnolvnlaobmrohkguzrxclvln/Build/Products/Debug-iphoneos/TestIn  
jectionXcode.app  
2 iPhone Developer: donganyuan101@gmail.com (X015UEA3WK)
```

The editor's status bar at the bottom indicates the file is in 'LF UTF-8 Plain Text' mode and has '1 update'.

使用

注意点

- 如果继续报错提示多个证书，不知道用哪个的问题，请查看keychain中是否有重名的多个证书，删掉所有同名旧证书，保留最新的一个，再次执行就好了。

提纲

- 插件安装与删除
- 原理简述
- 使用
- 局限

局限

Objective-C

- 静态变量
- 静态函数
- 全局函数

Swift

- 静态变量
- 静态函数
- 全局函数
- 非final或private的class
- struct, enum

局限

- 当注入代码中包含单例，当新class注入时，会重新生成一个新单例对象。所以，以shared开头的类方法名，是插件不会swizzle该方法。

局限

- 全局函数不会被注入，因为全局函数是静态链接到应用的。但是，我们可以在class中添加一个代理函数，去调用全局函数。

```
1
2 #import "Injectable.h"
3
4 void dispatch_on_main( void (^block)(void) ) {
5     dispatch_async( dispatch_get_main_queue(), block );
6 }
7
8 static id sharedInstance;
9
10 @implementation Injectable
11
12 + (instancetype)sharedInstance {
13     static dispatch_once_t once;
14     dispatch_once( &once, ^{
15         sharedInstance = [[self alloc] init];
16     } );
17     return sharedInstance;
18 }
19
20 - (void)injected {
21     NSLog( @"injected: %@", self );
22 }
23
24 - (void)doSomething {
25     dispatch_on_main( ^{ NSLog( @"main queue: %@", self ); } );
26 }
27
28 @end
29
```

参考资料

injection for Xcode

<https://github.com/johnno1962/injectionforxcode>

Dynamic linking on iOS

<http://ddeville.me/2014/04/>

About Loadable Bundles

https://developer.apple.com/library/mac/documentation/Cocoa/Conceptual>LoadingCode/Concepts/AboutLoadableBundles.html#//apple_ref/doc/uid/20001268-BCIDBAEJ