

一、安装签名工具ldid

- 先确保安装了[brew](#)

```
$ /usr/bin/ruby -e "$(curl -fsSL  
https://raw.githubusercontent.com/Homebrew/install/master/install)"
```

- 利用brew安装ldid

```
$ brew install ldid
```

二、修改环境变量

- 编辑用户的配置文件

```
$ vim ~/.bash_profile
```

- 在.bash_profile文件后面加入以下2行

```
export THEOS=~/.theos  
export PATH=$THEOS/bin:$PATH
```

- 让.bash_profile配置的环境变量立即生效（或者重新打开终端）

```
$ source ~/.bash_profile
```

三、下载theos

- 建议在\$THEOS目录下载代码（也就是刚才配置的~/.theos目录）

```
$ git clone --recursive https://github.com/theos/theos.git $THEOS
```

四、新建tweak项目

- cd到一个存放项目代码的文件夹（比如桌面）

```
$ cd ~/Desktop  
$ nic.pl
```

- 选择[11.] iphone/tweak

```
~/Desktop mj$ nic.pl  
NIC 2.0 - New Instance Creator  
-----  
[1.] iphone/activator_event  
[2.] iphone/application_modern  
[3.] iphone/cydget  
[4.] iphone/flipswitch_switch  
[5.] iphone/framework  
[6.] iphone/ios7_notification_center_widget  
[7.] iphone/library  
[8.] iphone/notification_center_widget  
[9.] iphone/preference_bundle_modern  
[10.] iphone/tool  
[11.] iphone/tweak  
[12.] iphone/xpc_service  
Choose a Template (required): 11
```

- 填写项目信息
 - Project Name
 - 项目名称
 - Package Name
 - 项目ID（随便写）
 - Author/Maintainer Name
 - 作者
 - 直接敲回车按照默认做法就行（默认是Mac上的用户名）
 - [iphone/tweak] MobileSubstrate Bundle filter
 - 需要修改的APP的Bundle Identifier（喜马拉雅FM的是com.gemd.iting）
 - 可以通过Cycrypt查看APP的Bundle Identifier
 - [iphone/tweak] List of applications to terminate upon installation
 - 直接敲回车按照默认做法就行

```
Project Name (required): ting_tweak
Package Name [com.yourcompany.ting_tweak]: com.mj.ting
Author/Maintainer Name [MJ Lee]:
[iphone/tweak] MobileSubstrate Bundle filter [com.apple.springboard]:
com.gemd.iting
[iphone/tweak] List of applications to terminate upon installation (space-
separated, '-' for none) [SpringBoard]:
Instantiating iphone/tweak in ting_tweak/...
Done.
```

五、编辑Makefile

- 在前面加入环境变量，写清楚通过哪个IP和端口访问手机
 - THEOS_DEVICE_IP
 - THEOS_DEVICE_PORT

```
export THEOS_DEVICE_IP=127.0.0.1
export THEOS_DEVICE_PORT=10010

include $(THEOS)/makefiles/common.mk

TWEAK_NAME = ting_tweak
ting_tweak_FILES = Tweak.xm

include $(THEOS_MAKE_PATH)/tweak.mk

after-install::
    install.exec "killall -9 SpringBoard"
```

- 如果不希望每个项目的Makefile都编写IP和端口环境变量，也可以添加到用户配置文件中
 - 编辑完毕后，`$ source ~/.bash_profile`让配置生效（或者重启终端）

```
$ vim ~/.bash_profile

export THEOS=~/.theos
export PATH=$THEOS/bin:$PATH
export THEOS_DEVICE_IP=127.0.0.1
export THEOS_DEVICE_PORT=10010

$ source ~/.bash_profile
```

六、编写代码

- 打开Tweak.xml文件

```
%hook XMAdAnimationView

- (id)initWithImageUrl:(id)arg1 title:(id)arg2 iconType:(long long)arg3
jumpType:(long long)arg4
{
    return nil;
}

%end

%hook XMSPatchPosterView

- (id)initWithFrame:(struct CGRect)arg1
{
    return nil;
}

%end
```

七、编译-打包-安装

- 编译

```
make
```

- 打包成deb

```
make package
```

- 安装（默认会自动重启SpringBoard）

```
make install
```

八、可能遇到的问题

1 - make package的错误

```
$ make package
```

```
Can't locate IO/Compress/Lzma.pm in @INC (you may need to install the
IO::Compress::Lzma module) (@INC contains: /Library/Perl/5.18/darwin-
thread-multi-2level /Library/Perl/5.18 /Network/Library/Perl/5.18/darwin-
thread-multi-2level /Network/Library/Perl/5.18 /Library/Perl/Updates/5.18.2
/System/Library/Perl/5.18/darwin-thread-multi-2level
/System/Library/Perl/5.18 /System/Library/Perl/Extras/5.18/darwin-thread-
multi-2level /System/Library/Perl/Extras/5.18 .) at
/Users/mj/theos/bin/dm.pl line 12.
BEGIN failed--compilation aborted at /Users/mj/theos/bin/dm.pl line 12.
make: *** [internal-package] Error 2
```

- 是因为打包压缩方式有问题，改成gzip压缩就行
 - 修改dm.pl文件，用#号注释掉下面两句

```
$ vim $THEOS/vendor/dm.pl/dm.pl

#use IO::Compress::Lzma;
#use IO::Compress::Xz;
```

- 修改deb.mk文件第6行的压缩方式为gzip

```
$ vim $THEOS/makefiles/package/deb.mk

_THEOS_PLATFORM_DPKG_DEB_COMPRESSION ?= gzip
```

2 - make的错误

```
$ make
```

```
Error: You do not have an SDK in
/Library/Developer/CommandLineTools/Platforms/iPhoneOS.platform/Developer/S
DKs
```

- 是因为多个xcode导致路径（有可能安装了好几个Xcode），需要指定一下Xcode

```
$ sudo xcode-select --switch
/Applications/Xcode.app/Contents/Developer/
```

```
$ make
```

```
> Making all for tweak xxx...
```

```
make[2]: Nothing to be done for `internal-library-compile'.
```

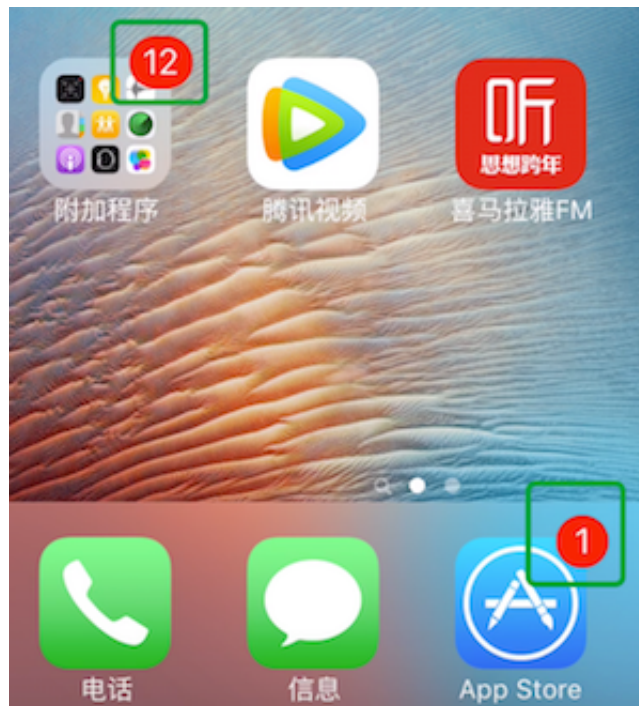
- 是因为之前已经编译过，有缓存导致的，clean一下即可

```
$ make clean
```

```
$ make
```

九、练习

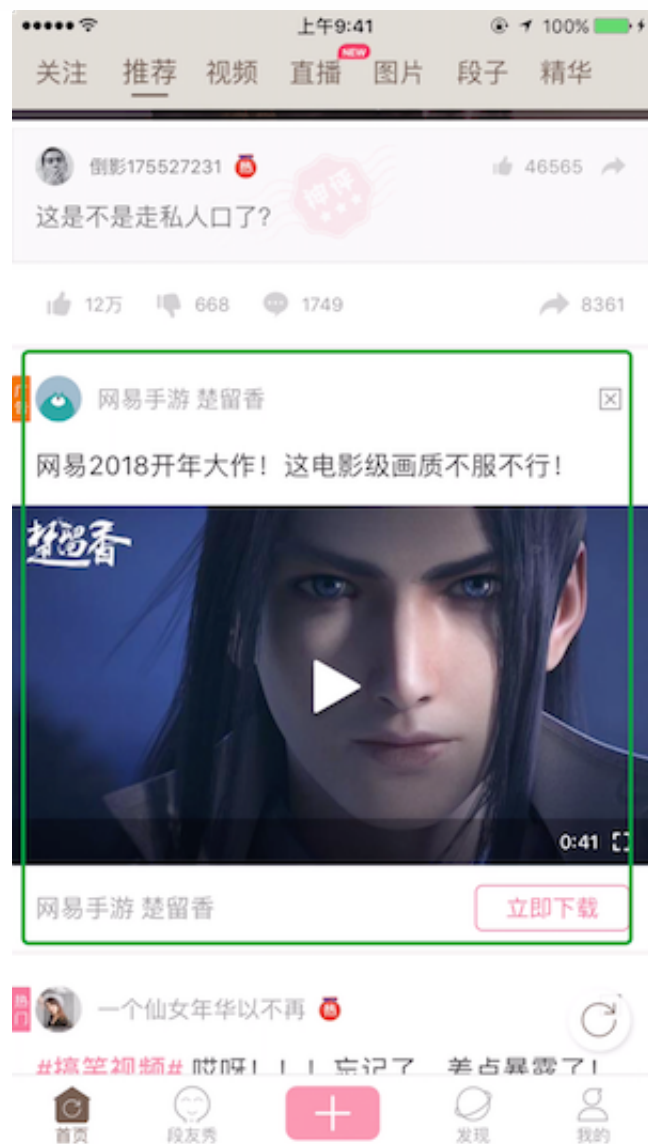
- 将桌面的更新数字去掉



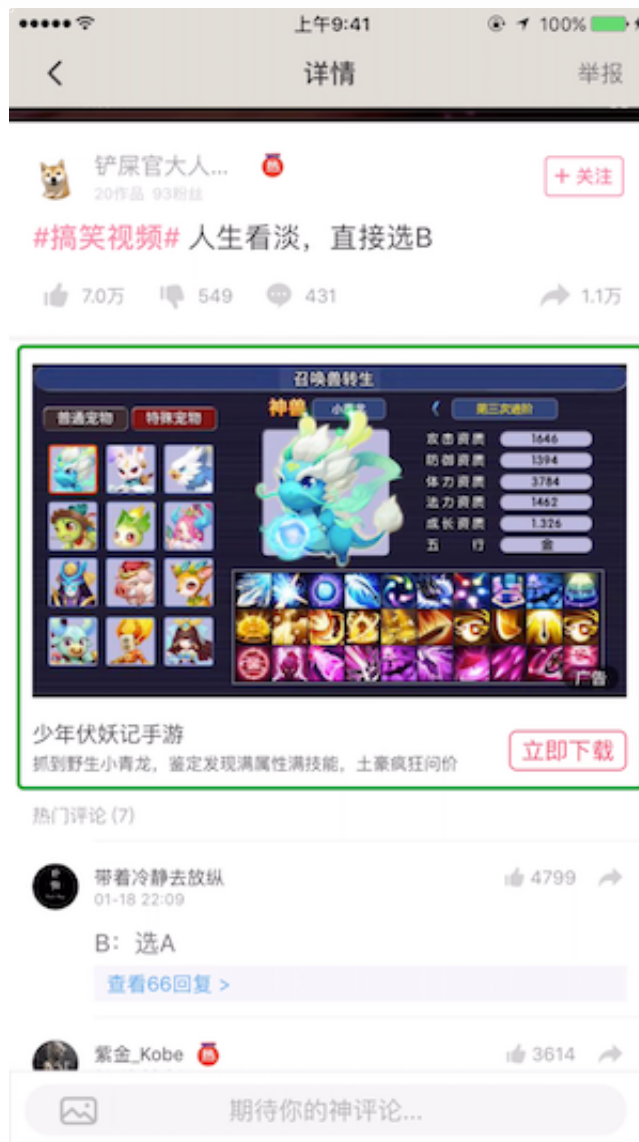
- 给微信的“发现”界面增加2行功能



- 去掉内涵段子的广告
 - 首页的广告



- 评论区的广告



十、theos资料查询

- 目录结构: <https://github.com/theos/theos/wiki/Structure>
- 环境变量: <http://iphonedevwiki.net/index.php/Theos>
- Logos语法: <http://iphonedevwiki.net/index.php/Logos>
 - **%hook**、**%end**: hook一个类的开始和结束
 - **%log**: 打印方法调用详情
 - 可以通过Xcode -> Window -> Devices and Simulators查看日志
 - **HBDebugLog**: 跟NSLog类似
 - **%new**: 添加一个新的方法
 - **%c(className)**: 生成一个Class对象, 比如%c(NSObject), 类似于NSStringFromClass()、objc_getClass()
 - **%orig**: 函数原来的代码逻辑
 - **%ctor**: 在加载动态库时调用

- **%dtor**：在程序退出时调用
- **logify.pl**：可以将一个头文件快速转换成已经包含打印信息的xm文件

```
logify.pl xx.h > xx.xm
```

- 如果有额外的资源文件（比如图片），放在项目的**layout**文件夹中，对应着手机的根路径/

十一、theos-tweak的实现过程

- 编写Tweak代码
- **\$ make**：编译Tweak代码为动态库 (*.dylib)
- **\$ make package**：将dylib打包为deb文件
- **\$ make install**：将deb文件传送到手机上，通过Cydia安装deb
- 插件将会安装在/Library/MobileSubstrate/DynamicLibraries文件夹中
 - *.dylib：编译后的Tweak代码
 - *.plist：存放着需要hook的APP ID
- 当打开APP时
 - Cydia Substrate（Cydia已自动安装的插件）会让APP去加载对应的dylib
 - 修改APP内存中的代码逻辑，去执行dylib中的函数代码
- 所以，theos的tweak并不会对APP原来的可执行文件进行修改，仅仅是修改了内存中的代码逻辑
- 疑问
 - 未脱壳的APP是否支持tweak？
 - 支持，因为tweak是在内存中实现的，并没有修改.app包中的可执行文件
 - tweak效果是否永久性的？
 - 取决于tweak中用到的APP代码是否被修改过
 - 如果一旦更新APP，tweak会不会失效？
 - 取决于tweak中用到的APP代码是否被修改过
 - 未越狱的手机是否支持tweak？
 - 不支持
 - 能不能对Swift\C函数进行tweak？
 - 可以，方式跟OC不一样
 - 能不能对游戏项目进行tweak？
 - 可以
 - 但是游戏大多数是通过C++\C#编写的，而且类名、函数名会进行混淆操作

十二、logify.pl注意点

- logify.pl生成的xm文件，有很多时候是编译不通过的，需要进行一些处理
 - 删掉__weak
 - 删掉inout
 - 删掉协议，比如
 - 或者声明一下协议信息@protocol XXTestDelegate
 - 删掉- (void).cxx_destruct { %log; %orig; }
 - 删除HBLogDebug(@" = 0x%x", (unsigned int)r);
 - 替换类名为void，比如将**XXPerson ***替换为**void ***
 - 或者声明一下类信息@class XXPerson