

iOS远程推送通知

在ios系统中，app应用程序无法在后台完成较多的任务，仅仅允许程序做一些有限的任务（如音视频播放、地理位置信息、voip）。然而，如果你想做一些有趣的事情，并且告知用户，甚至用户没有使用你的app应用程序，如收到一条微博信息，自己心爱的球队夺冠又或者晚餐已经准备好了等信息时，由于你的app应用程序没有运行（或者程序进入后台，进程被挂起），因此，可能无法接收到相关的消息。

然而，苹果提供了这类情况的解决方案，替代你的app应用程序定时获取事件或者在后台去处理这些事件，你需要搭建服务器端来处理这类事情。当感兴趣的事情发生时，服务器端会发送一条推送通知到你的app应用程序。推送通知可以做三件事情，1文字信息，2一种声音，3一个徽章的标记号。你可以将以上三种情况任意组合，这取决于你的app应用程序。比如推送通知时，只显示声音和标记，不显示消息等。

1、推送通知流程简述

- （1） 首先用户必须确认需要接收到这些消息；
- （2） app程序接收到令牌信息；
- （3） 将令牌信息发送到服务器端；
- （4） 当感兴趣的事情发生时，你的服务器将向苹果的推送通知服务器（Apple Push Notification Service，以下简称APNS）发送通知；
- （5） APNS会向你的设备发送消息，即通知；

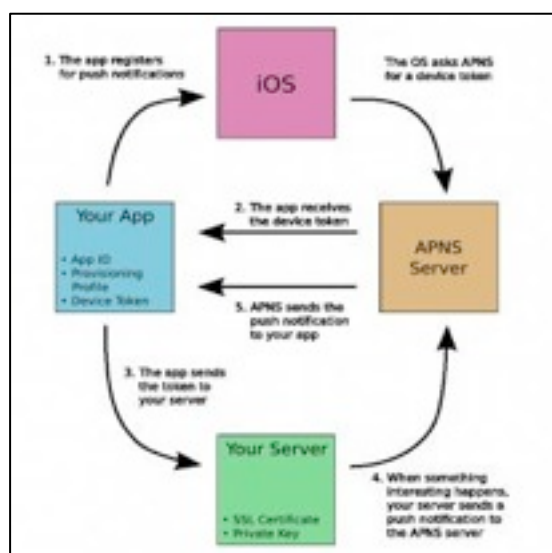


图1-1 推送通知流程图

2、推送通知基本要求

(1) 首先需要iphone或者ipad设备，模拟器上无法对“推送通知”进行测试。因此，你必须在真机上进行测试；

(2) 需要加入ios开发者计划（99美金/年）；

(3) 一台连接互联网的服务器，因为推送通知的发起者是你的服务器（provider）；

3、推送通知内容解析

服务器端（provider）负责创建消息推送通知，一个推送服务通知主要包含“设备令牌”和“负载”（定义的json信息的属性列表，负载最大运行值为256字节）。对于每个推送通知，服务器端（provider）必须构建一个严格遵守RFC 4627 的 JSON 字典。该字典必须包含一个键名为 aps 的字典。而 aps 的字典包含一个或多个属性，这些属性指定如下动作：

(1) 提示用户的警告信息（ios5后有所改变，但大体一致）；

(2) 一个提示声音；

(3) 显示app应用程序的数字（徽章）；

4、推送通知的局限性

它们是不可靠的！不同的设备之间，他们的推送消息的时间可能存在不一致。

推送通知也可能不会被送达到指定设备。原因是可能所在wifi环境下端口被封、设备处于关机状态，又或者网络环境较差，以至无法正确送达。APNS会尽量与设备“握手”以此建立连接，但它只会尝试在有限的时间，一旦超时，推送通知将永远消失。

5、推送通知步骤详述

消息推送通知步骤与真机调试应用程序（包括发布程序流程）流程基本相同。只是在创建App ID时，与后者有所差别。此外，还需要向服务器端提供pem文件（该文件用于服务器与APNS服务器建立TSL点对点连接）。

5.1、请求CSR 文件

单击“钥匙串访问”，选择“证书助理”一项，单击下拉菜单中“从证书颁发机构请求证书”项（如若没有该选择项，则需要下载“WWDR Intermediate Certificate”），此外需要确保没有选中“钥匙访问串”中任何密钥。如下图所示：



图5-1 请求CSR文件

5.2、制作CSR文件

填写用户邮件地址以及给钥匙命名，选择CSR文件存储路径（全称 Certificate Signing Request，用于向苹果请求开发证书和push证书），如下图所示：



图5-2 生成CSR文件

5.3、导出私钥

单击新生成的密钥（共含公钥和私钥），将私钥导出，输入导出密码，并牢记此密码，如下图所示：

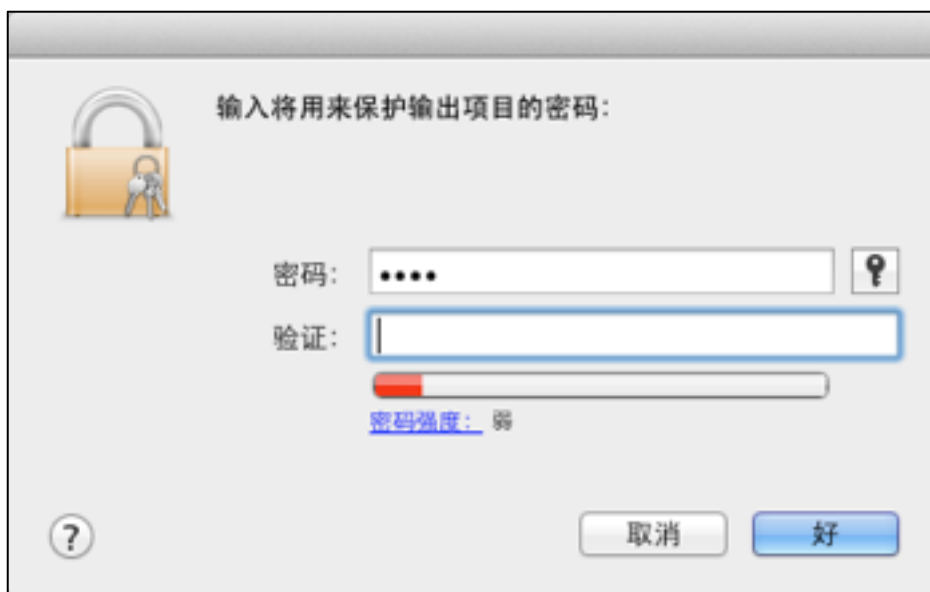


图5-3 导出私钥

5.4、制作开发证书

进入苹果IDP（ios Developer Program）页面，选择certificates一栏，请求开发证书。选择已生成好CSR文件，单击submit按钮，生成开发证书。（刷新页面后，可下载该证书）如下图所示：

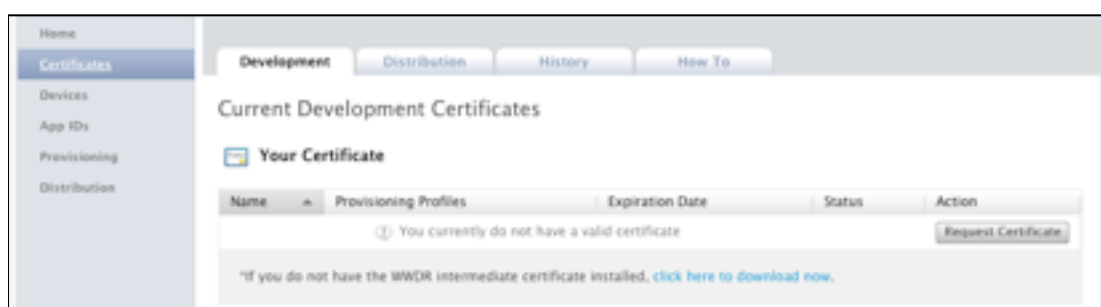


图5-4 生成开发者证书

5.5、安装开发证书

下载完成证书后（.cer文件），双击证书将开发证书添加至钥匙串访问中，如下图所示：

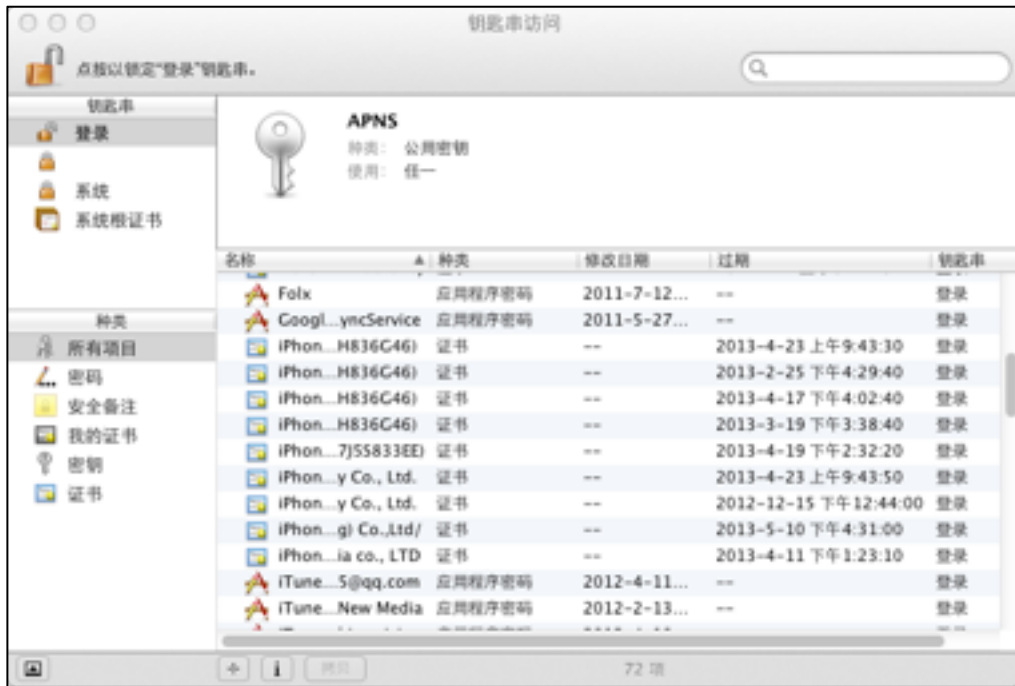


图5-5 安装开发者证书

5.6、创建App ID

添加完成开发证书后，选择App IDs一栏，创建一个新的应用程序ID。在“Bundle Identifier”项中，将绑定标识符设置为“com.yourcompany.pushApp”（固定写法）而不是“com.yourcompany.*”（每一个带有推送应用程序，需要拥有自己独特的ID，因为推送通知需要被发送到一个特定的应用中，因此，不可以使用通配符“*”）如下图所示：

Manage
How To

Create App ID

Description
Enter a common name or description of your App ID using alphanumeric characters. The description you specify will be used throughout the Provisioning Portal to identify this App ID.
 App ID Name
You cannot use special characters as @, &, ", " in your description.

Bundle Seed ID (App ID Prefix)
Your Team ID (PK53GA854L) will be used as the App ID Prefix.

Bundle Identifier (App ID Suffix)
Enter a unique identifier for your App ID. The recommended practice is to use a reverse-domain name style string for the Bundle Identifier portion of the App ID.
 com.yourCompany.pushApp
Example: com.domainname.appname

Cancel
Submit

图5-6 创建App ID

5.7、制作push SSL证书

App ID生成后，下一步将配置App ID。单击“configure”，选中“Enable for Apple Push Notification service”，“push SSL certificate”项即可被激活。选择配置“Development Push SSL Certificate”项（“production Push SSL certificate”证书用于该产品发布时，“Development Push SSL Certificate”用于程序测试），单击“configure”根据向导生成“push SSL”证书。进入到“submit certificate signing request”页时，再次选择已生成CSR文件，单击submit按钮生成push SSL证书。点击“download”按钮下载该证书，双击后添加至“钥匙串访问中”，如下图所示：

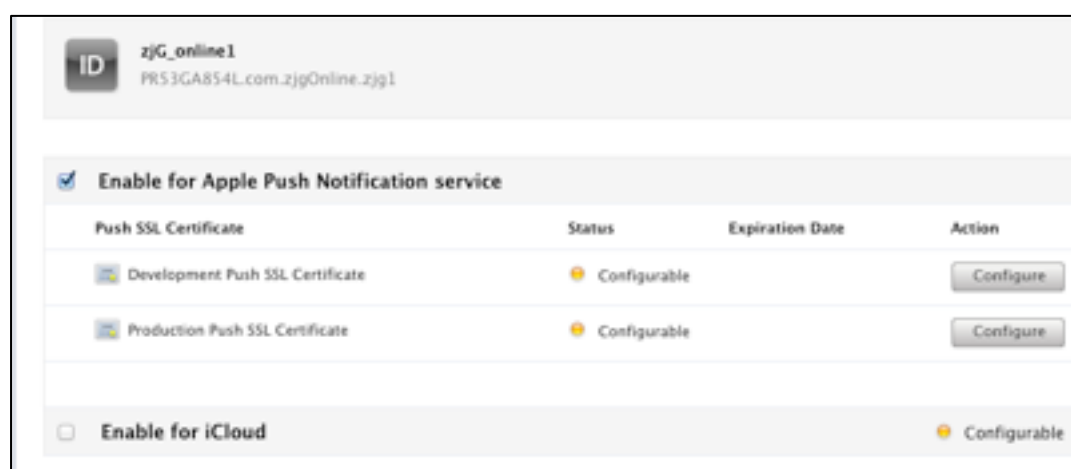


图5-7-1 配置app ID

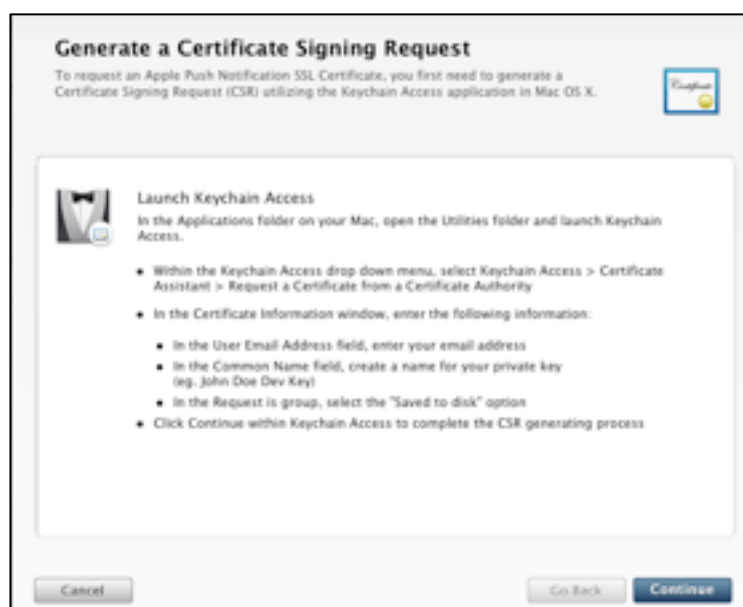


图5-7-2 ssl证书向导页



图5-7-3 选择CSR文件



图5-7-4 生成ssl证书

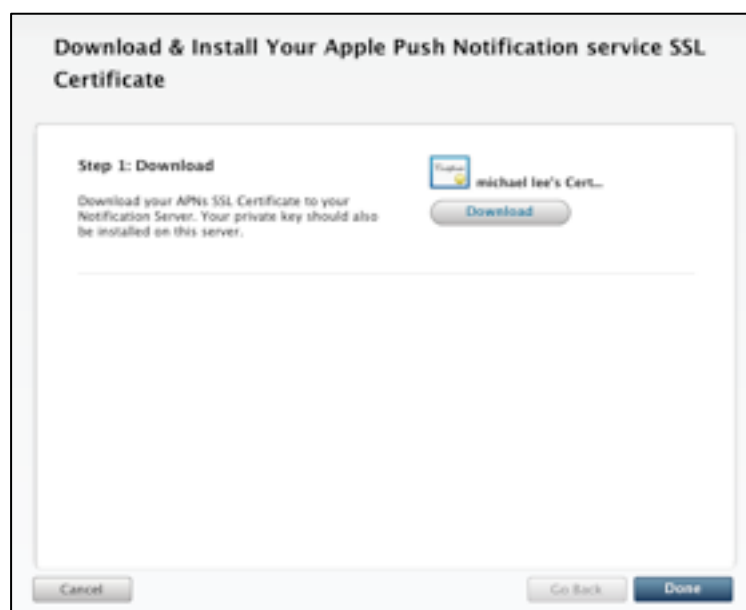


图5-7-5 下载ssl证书

5.8、制作provisioning文件

填写该文件名称，选择签发证书机构，再选择已生成App ID，最后一步选择需要测试的设备。单击“submit”生成provisioning文件，下载后，双击安装。如下图所示：

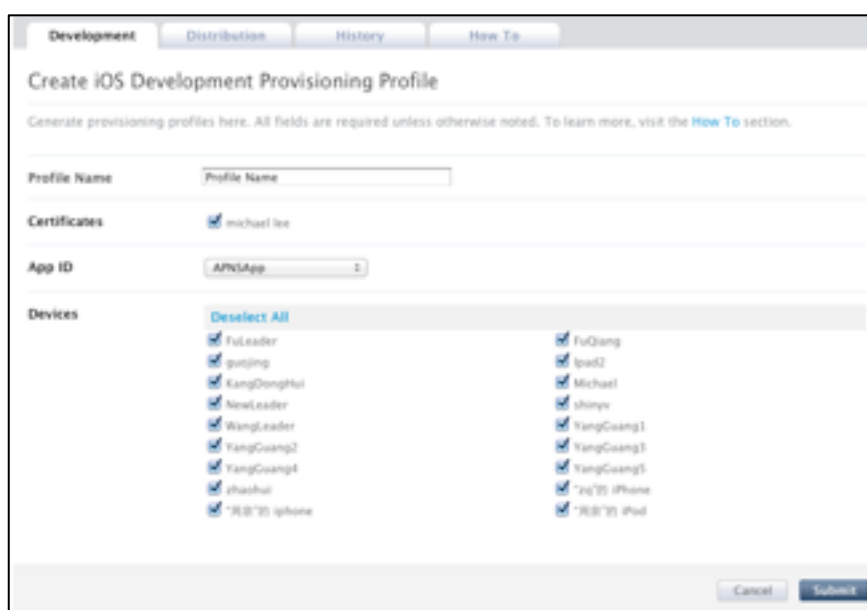


图5-8 生成ssl证书

5.9、制作PEM文件

PEM文件由私钥和push SSL证书合成的文件。当服务器端（provider）与APNS服务器建立连接时，需要对该文件进行验证。

（1）准备私钥和push SSL证书

私钥是在5.3步骤中获得到，push SSL证书是在步骤5.7中得到。

(2) 将证书制作成.pem文件

打开终端应用程序，访问到push SSL目录下（如访问到桌面APNS文件夹下，在终端输入“`cd /Users/userName/Desktop/APNS`”），输入命令“`openssl x509 -in aps_developer_identity.cer -inform der -out push_developer_cer.pem`”（将.cer文件“aps_developer_identity”转换成.pem文件“push_developer_cer”）。运行命令后，在该文件夹产生一个新的文件“push_developer_cer.pem”。

(3) 将私钥制作成.pem文件

在终端程序中输入“`openssl pkcs12 -nocerts -out pushKey.pem -in TestPushKey.p12`”命令，输入该命令后，终端程序提示输入密钥密码，键入密码后，终端继续提示输入新密码以及确认密码步骤。成功验证后产生新的文件“pushKey.pem”。

(4) 合并证书和私钥

将上述新产生的文件合并，在终端输入命令“`cat push_developer_cer.pem pushKey.pem > ck.pem`”，完成后生成“ck.pem”文件。该文件在服务器与APNS服务器建立连接时，用于验证。

(5) 测试证书是否有效

测试证书是否有效，在终端中输入“`openssl s_client -connect gateway.sandbox.push.apple.com:2195`

`-cert push_developer_cer.pem -key pushKey.pem`”命令。如若测试成功，会显示大量openssl消息。当你任意键入几个字符时，服务器会提示断开。（切勿复制）

附：

服务器端测试用例（PHP）

```
<?php
// Put your device token here (without spaces):
$deviceToken =
'0f744707bebcf74f9b7c25d48e3358945f6aa01da5ddb387462c7eaf61bbad78';

// Put your private key's passphrase here:
$passphrase = 'pushchat';

// Put your alert message here:
$message = 'My first push notification!';
////////////////////////////////////

$ctx = stream_context_create();
stream_context_set_option($ctx, 'ssl', 'local_cert', 'ck.pem');
stream_context_set_option($ctx, 'ssl', 'passphrase', $passphrase);

// Open a connection to the APNS server
$fp = stream_socket_client(
    'ssl://gateway.sandbox.push.apple.com:2195', $err,
    $errstr, 60, STREAM_CLIENT_CONNECT|
    STREAM_CLIENT_PERSISTENT, $ctx);

if (!$fp)
    exit("Failed to connect: $err $errstr" . PHP_EOL);

echo 'Connected to APNS' . PHP_EOL;

// Create the payload body
$body['aps'] = array(
    'alert' => $message,
    'sound' => 'default'
```

```
);

// Encode the payload as JSON
$payload = json_encode($body);

// Build the binary notification
$msg = chr(0) . pack('n', 32) . pack('H*', $deviceToken) . pack('n', strlen($payload)) .
$payload;

// Send it to the server
$result = fwrite($fp, $msg, strlen($msg));

if (!$result)
    echo 'Message not delivered' . PHP_EOL;
else
    echo 'Message successfully delivered' . PHP_EOL;

// Close the connection to the server
fclose($fp);
```

mac苹果系统测试用例

地址: <https://github.com/stefanhafeneger/PushMeBaby> (直接将工程中
aps_developer_identity文件替换成自己的push SSL证书, 将设备令牌写入初
始化方法)

参考文章:

http://developer.apple.com/library/ios/#documentation/NetworkingInternet/Conceptual/RemoteNotificationsPG/ApplePushService/ApplePushService.html#//apple_ref/doc/uid/TP40008194-CH100-SW1

<http://www.cocoachina.com/bbs/read.php?tid=20723&keyword=apns>

<http://www.raywenderlich.com/3443/apple-push-notification-services-tutorial-part-12>

<http://www.cocoachina.com/newbie/basic/2010/0401/900.html>

<http://linwww.iteye.com/blog/1433189>