

Push Notification IOS

“PHP code”

2015.08.07

Agenda

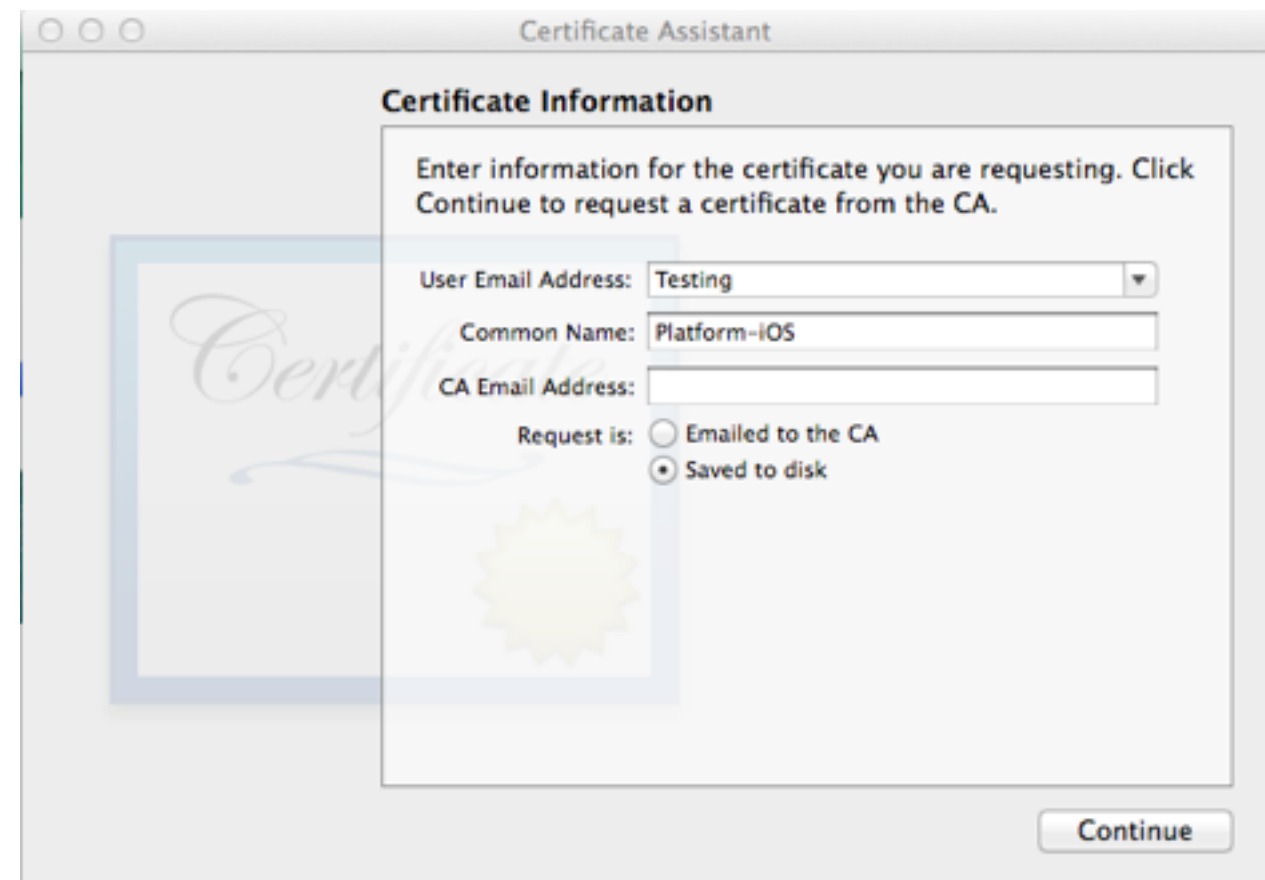
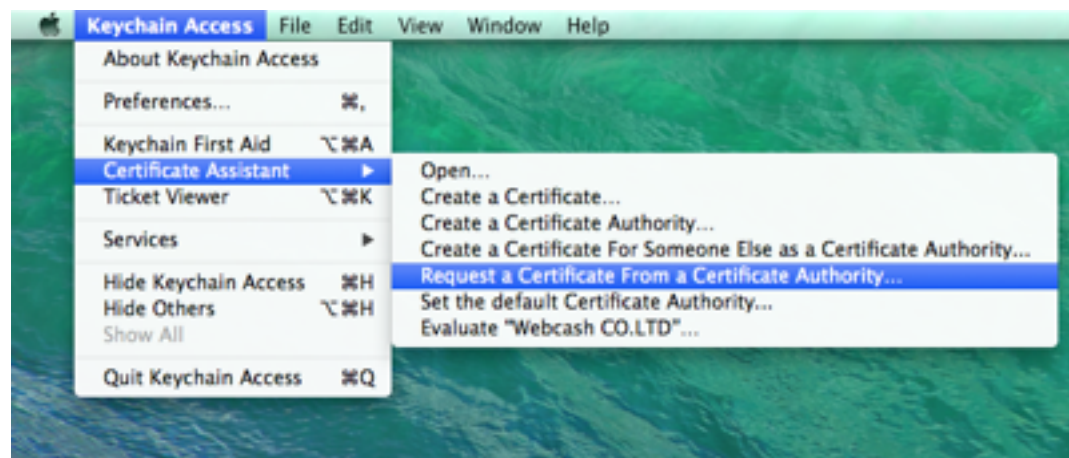
1. Creating the SSL certificate
 - 1.1 Generating a Certificate Request
 - 1.2 Creating an App ID
 - 1.3 Configuring your App ID for Development Push Notifications
2. Creating the Provisioning Profile
3. Convert file .p12 to .pem
4. Configuring Xcode
5. Process of PHP code
6. Process of APNs
7. Example
8. Note

1. Creating the SSL certificate

1.1 Generating a Certificate Request

To begin, we'll need a certificate signing request file. This will be used to authenticate the creation of the SSL certificate.

- Launch the **Keychain Access** application on your Mac.
- Select the menu item Keychain Access > Certificate Assistant > Request a Certificate From a Certificate Authority...
- Enter your email address and name.
- Select "Saved to disk" to download the .certSigningRequest file to your desktop.



1.2 Creating an App ID

- Navigate to the **Apple Developer Member Center website**, and select **Certificates, Identifiers & Profiles**.
- Select **Identifiers** from the iOS Apps section.
- You will see a list of your iOS App IDs. Select the + button to register a new App Id.

Register iOS App ID

ID Registering an App ID

The App ID string contains two parts separated by a period (.)—an App ID Prefix that is defined as your Team ID by default and an App ID Suffix that is defined as a Bundle ID search string. Each part of an App ID has different and important uses for your app. [Learn More](#)

App ID Description

Name:

You cannot use special characters such as @, &, %, ' , " , *

App ID Prefix

Value:

App ID Suffix

☒ **Explicit App ID**

If you plan to incorporate app services such as Game Center, In-App Purchase, Data Protection, and iCloud, or want a provisioning profile unique to a single app, you must register an explicit App ID for your app.

To create an explicit App ID, enter a unique string in the Bundle ID field. This string should match the Bundle ID of your app.

Bundle ID:

We recommend using a reverse-domain name style string (i.e., com.domainname.appname). It cannot contain an asterisk (*).

☐ **Wildcard App ID**

This allows you to use a single App ID to match multiple apps. To create a wildcard App ID, enter an asterisk (*) as the last digit in the Bundle ID field.

Bundle ID:

Example: com.domainname.*

App Services

Select the services you would like to enable in your app. You can edit your choices after this App ID has been registered.

Enable Services:

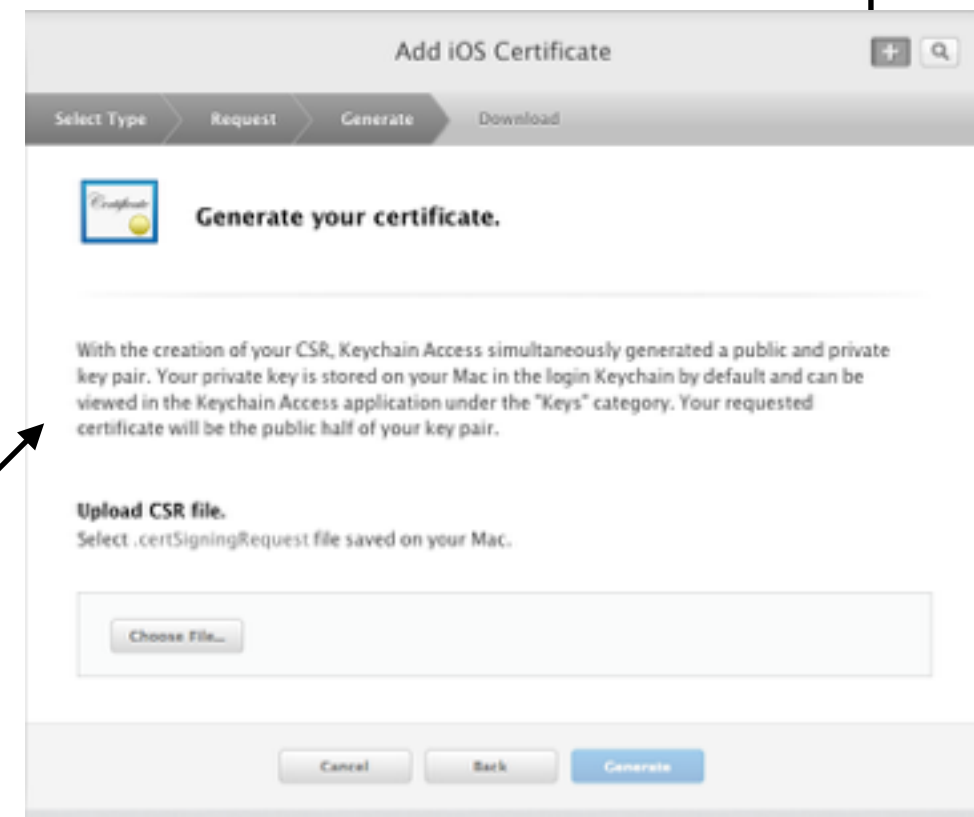
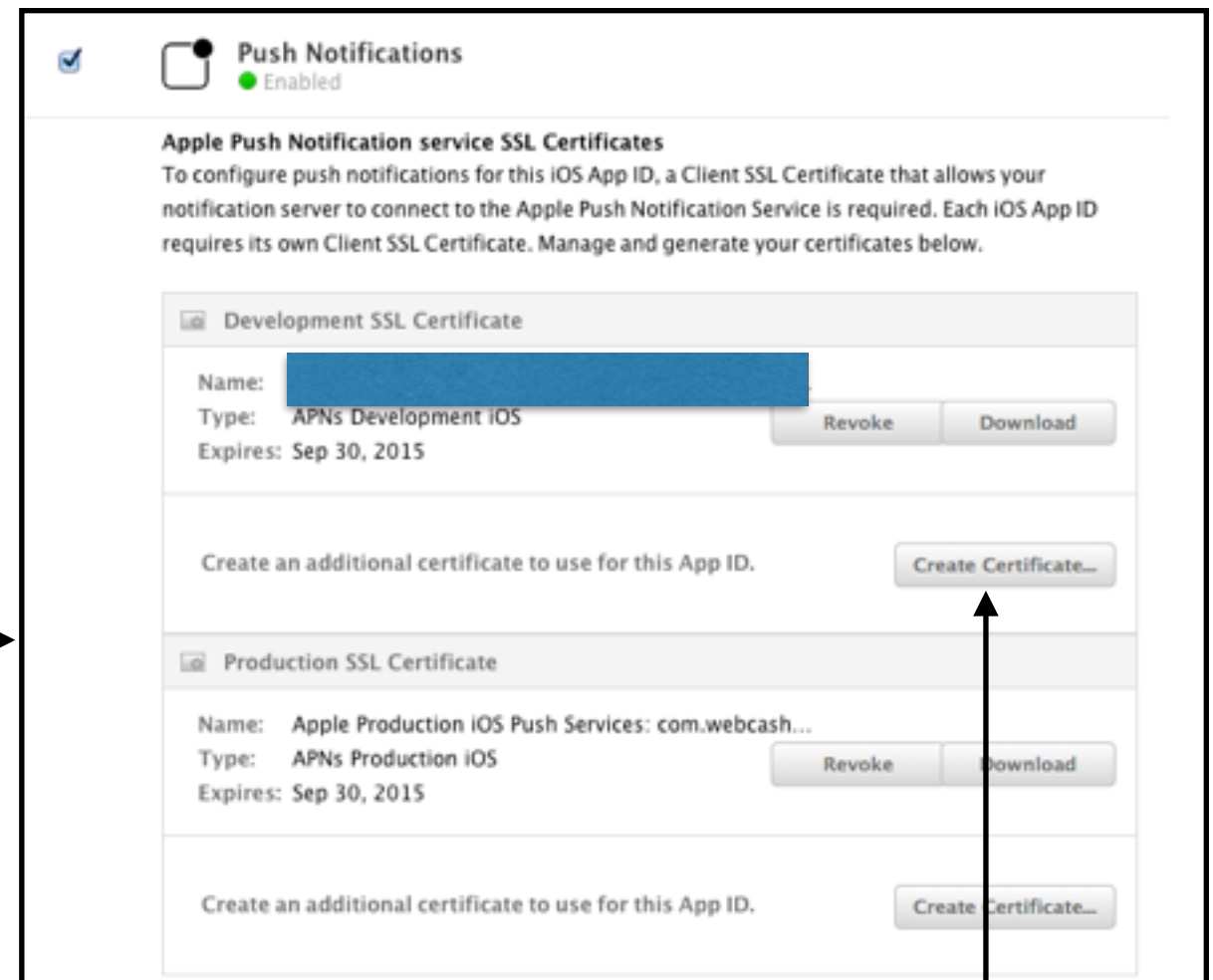
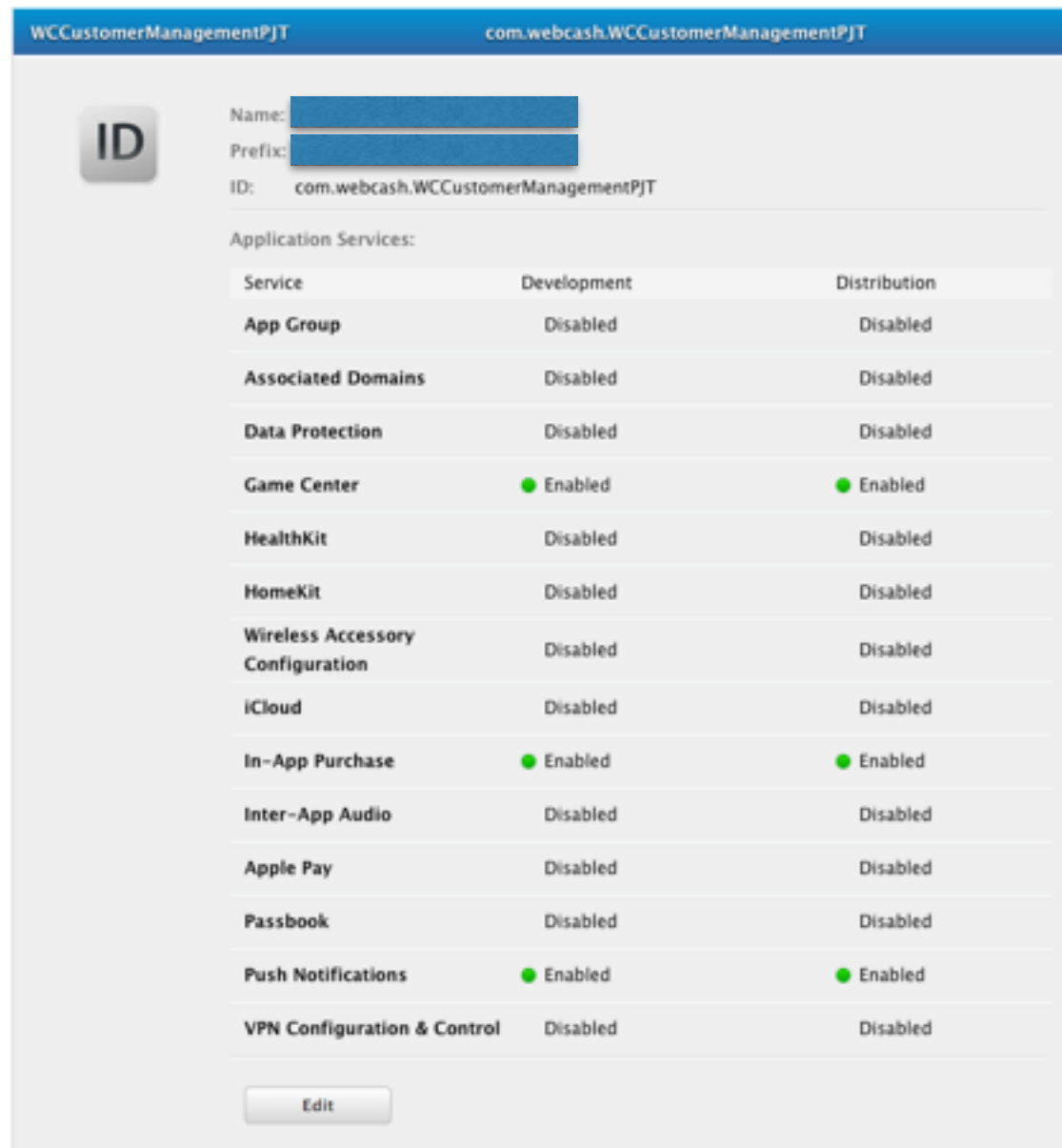
- ☐ App Groups
- ☐ Associated Domains
- ☐ Data Protection
 - ☐ Complete Protection
 - ☐ Protected Unless Open
 - ☐ Protected Until First User Authentication
- ☒ Game Center
- ☐ HealthKit
- ☐ HomeKit
- ☐ Wireless Accessory Configuration
- ☐ Apple Pay (Beta)
- ☐ iCloud
 - ☐ Compatible with Xcode 5
 - ☐ Include CloudKit support (requires Xcode 6)
- ☒ In-App Purchase
- ☐ Inter-App Audio
- ☐ Passbook
- ☒ Push Notifications
- ☐ VPN Configuration & Control

Cancel Continue

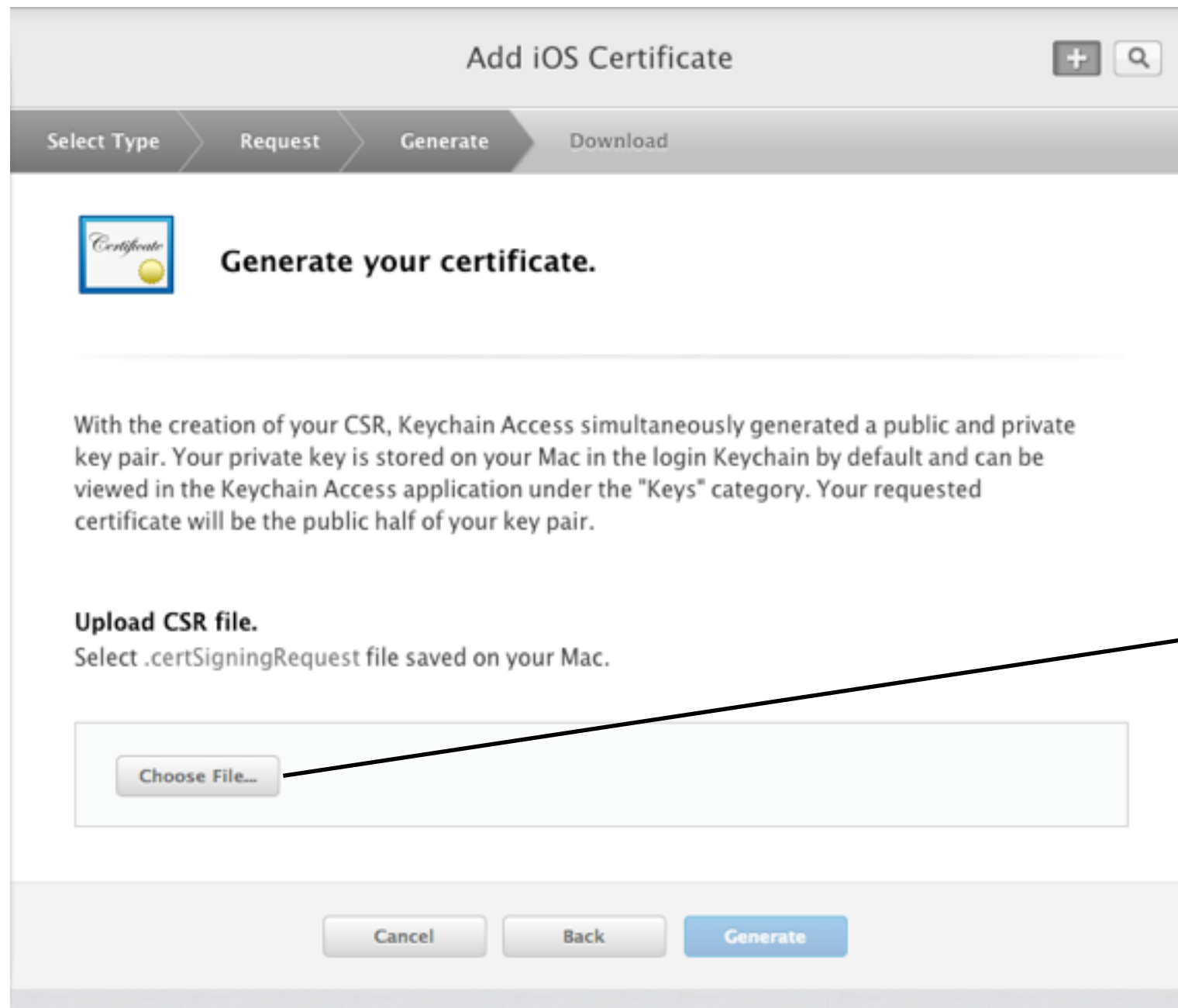
allow push notification

1.3 Configuring your App ID for Development Push Notifications


Select your newly created App ID from the list of iOS App IDs, then select "Edit".



you don't have certificate yet go and create new one



choose this file

 CertificateSigningRequest.certSigningRequest

Select "Generate". Once the certificate is ready, select "Done" and download the generated SSL certificate from the "iOS App ID Settings" screen

Certificates, Identifiers & Profiles

Lay Vitou ▾

iOS Apps ▾

✓ Certificates

- ☐ All
- ☐ Pending
- ☐ Development
- ☐ Production

ID Identifiers

- ☐ App IDs
- ☐ Pass Type IDs
- ☐ Website Push IDs
- ☐ iCloud Containers
- ☐ App Groups
- ☐ Merchant IDs

📱 Devices

- ☐ All

📄 Provisioning Profiles

- ☐ All
- ☐ Development
- ☐ Distribution

Add iOS Certificate



Select Type

Request

Generate

Download



Your certificate is ready.

Download, Install and Backup

Download your certificate to your Mac, then double click the .cer file to install in Keychain Access. Make sure to save a backup copy of your private and public keys somewhere secure.



Name: Apple Development iOS Push Services: com.web[redacted]
 Type: APNs Development iOS
 Identifier ID: WCCu[redacted]
 Expires: Sep 30, 2015

[Download](#)

click to download

Documentation

For more information on using and managing your certificates read:

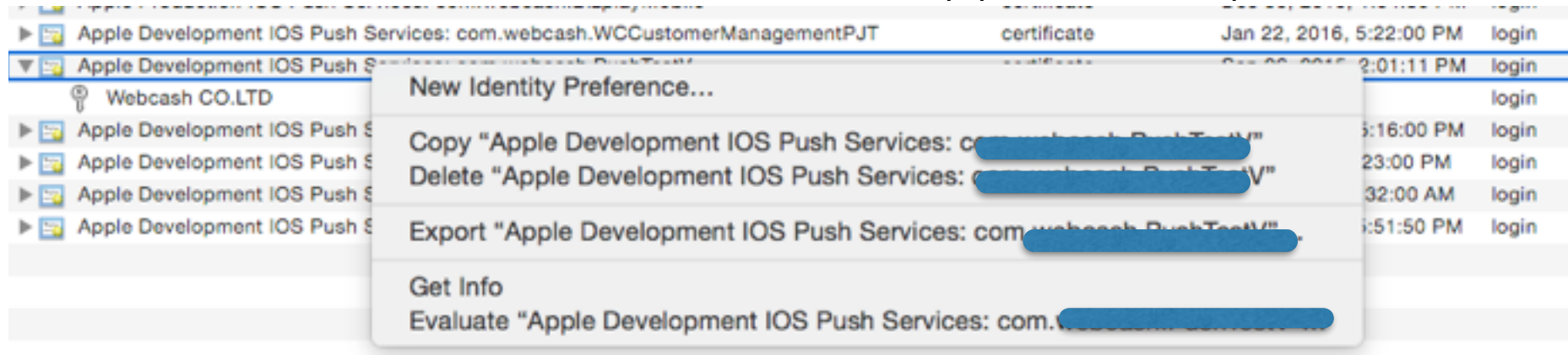
[App Distribution Guide](#)
[Add Another](#)
[Done](#)

- Double click on the downloaded SSL certificate to install it in your Keychain.

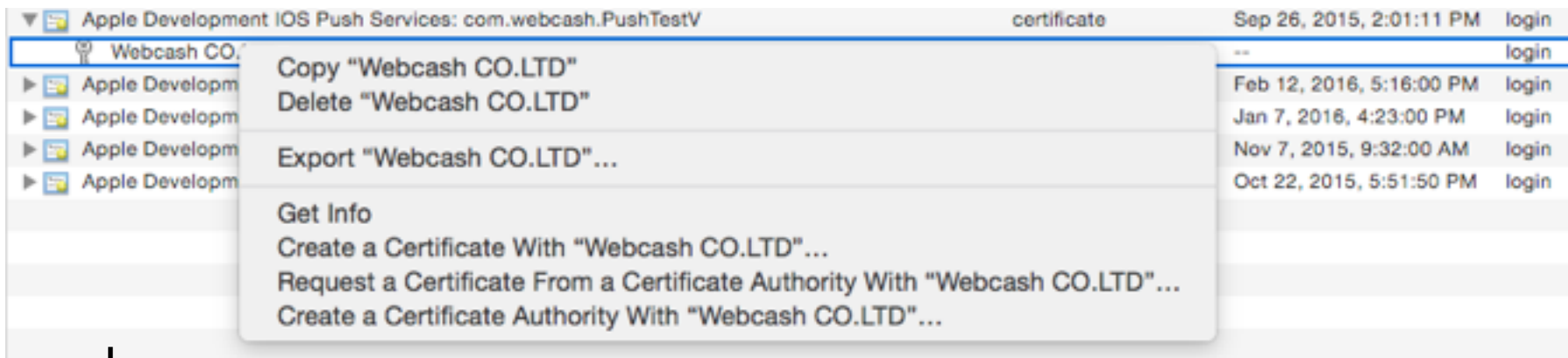
there are two .p12 that you will get:

- Certificate.p12
- key.p12

click certificate that u download from apple developer site and export



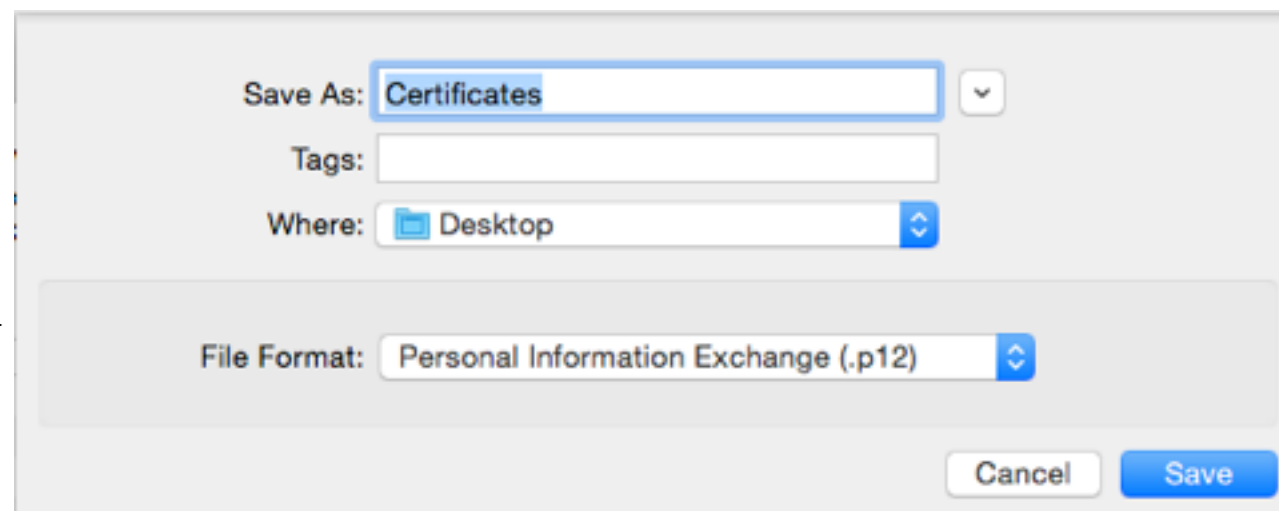
click key and export



as name key.p12



key.p12

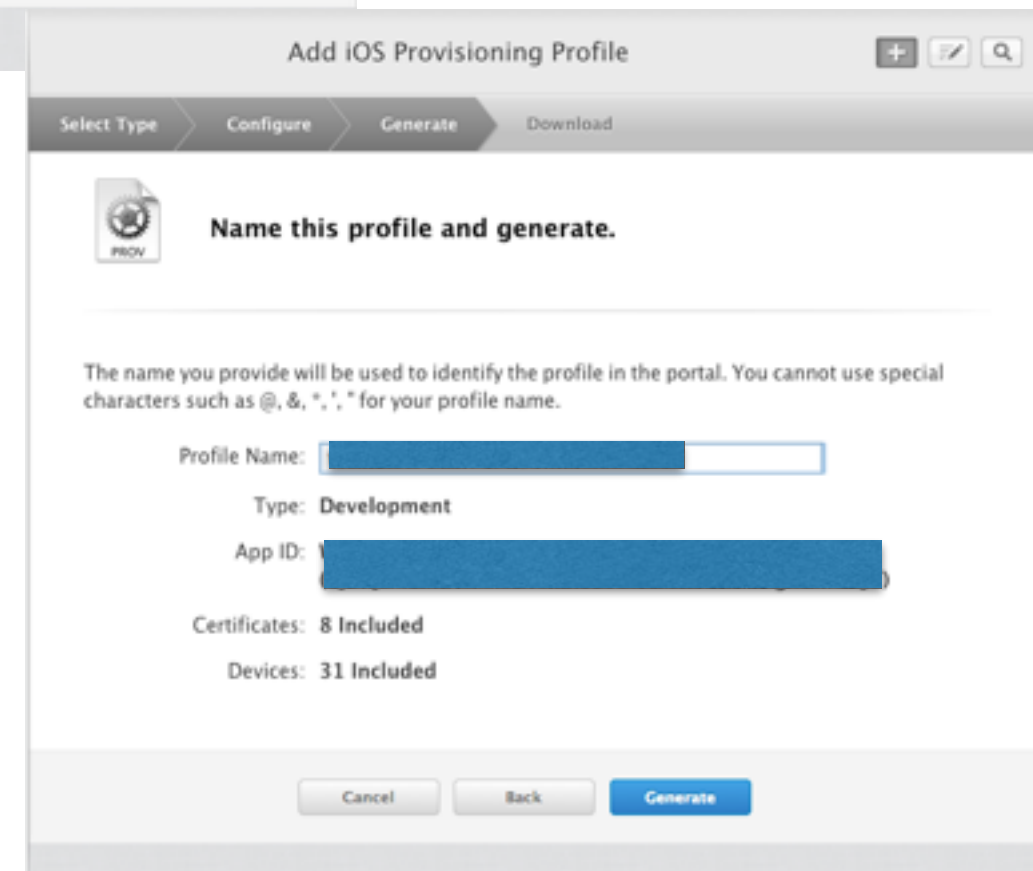
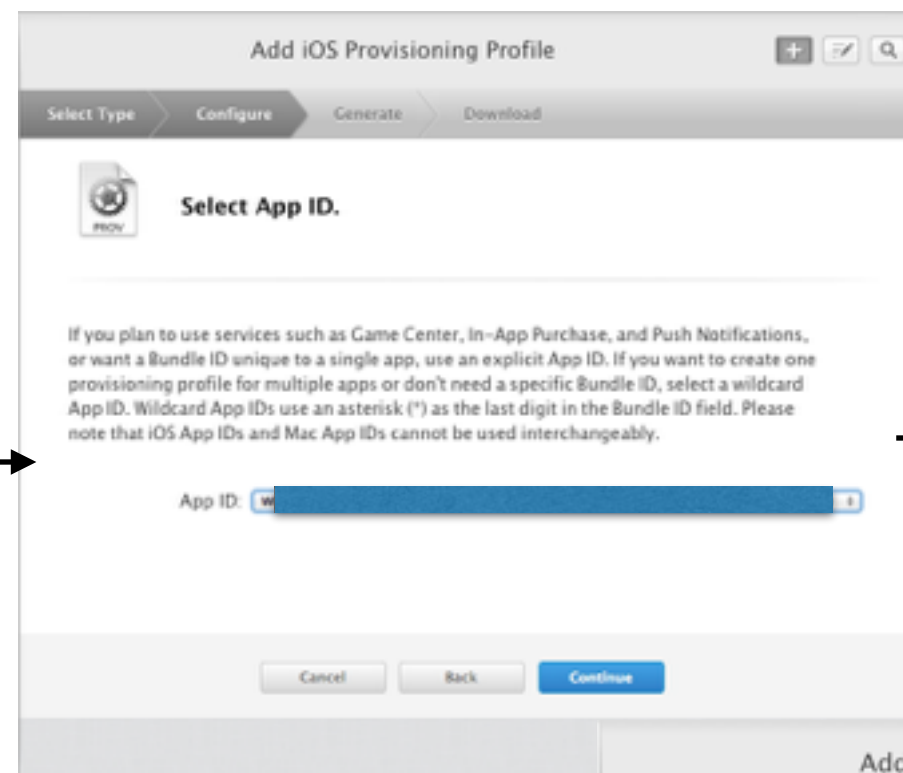


cert.p12

as name certificate.p12

2. Creating the Provisioning Profile

- Navigate to the Apple Developer Member Center website, and select Certificates, Identifiers & Profiles.
- Select Provisioning Profiles from the iOS Apps section.
- Select the + button to create a new iOS Provisioning Profile.
- Choose "iOS App Development" as your provisioning profile type then select "Continue". We will create Ad Hoc and App Store profiles later.



Download and double click to active



choose your Provision that double click

PROJECT

ApplePush

TARGETS

ApplePush

General

Capabilities

Info

Build Settings

Build Phases

Build Rules

Basic

All

Combined

Levels

+

Q~

Debug Information Format	DWARF with dSYM File
Embedded Content Contains Swift Code	No
Generate Profiling Code	No
Precompiled Header Uses Files From Build Directory	Yes
Require Only App-Extension-Safe API	No
Scan All Source Files for Includes	No
Validate Built Product	<Multiple values>
Debug	No
Release	Yes

Code Signing

Setting

ApplePush

Code Signing Entitlements

Code Signing Identity

Debug

Any iOS SDK

Release

Any iOS SDK

Code Signing Resource Rules Path

Other Code Signing Flags

Provisioning Profile

Debug

Any iOS SDK

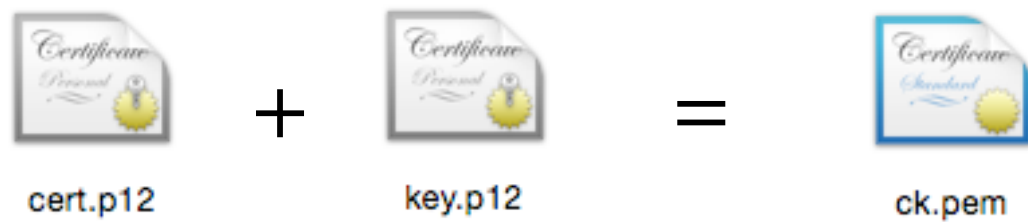
Release

Any iOS SDK

Any iOS SDK

Any iOS SDK

3. Convert file .p12 to .pem




Go and open Terminal and type cmt below :

```
openssl pkcs12 -clcerts -nokeys -out cer.pem -in cert.p12
```

```
openssl pkcs12 -nocerts -out key.pem -in key.p12
```

```
openssl rsa -in key.pem -out key.unencrypted.pem
```

 this one for clear pwd from key.pem

```
cat cer.pem key.unencrypted.pem > ck.pem
```

```
smart-imac:Desktop Yoman$ openssl pkcs12 -clcerts -nokeys -out cer.pem -in cert.p12
Enter Import Password:
MAC verified OK
```

 just enter no need to put pwd

- but for key.p12 u need to put pwd
- the last one is margin that 2 file in one file call ck.pem

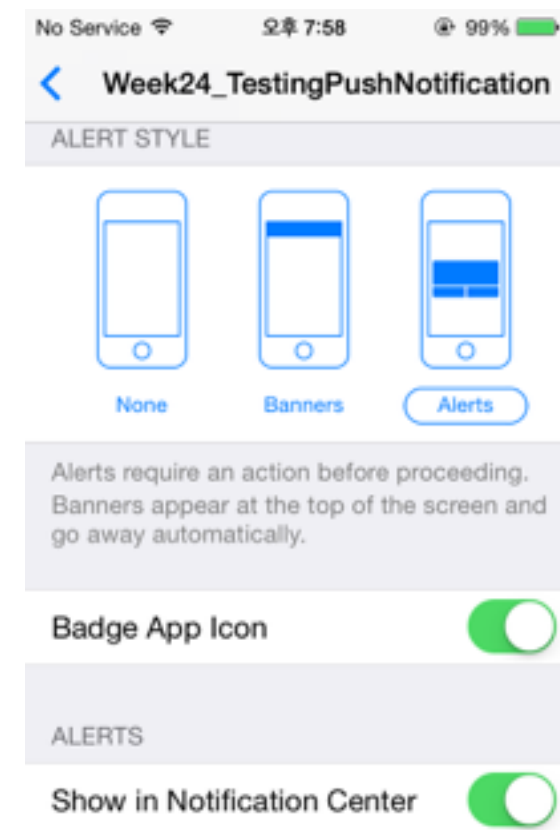
4. Configuring Xcode

AppDelegate.m

```
- (BOOL)application:(UIApplication *)application didFinishLaunchingWithOptions:(NSDictionary *)launchOptions
{
    // Let the device know we want to receive push notifications
    [[UIApplication sharedApplication] registerForRemoteNotificationTypes:(UIRemoteNotificationTypeAlert |
        UIRemoteNotificationTypeBadge | UIRemoteNotificationTypeNone)];
    return YES;
}
```



Allow notification with your app



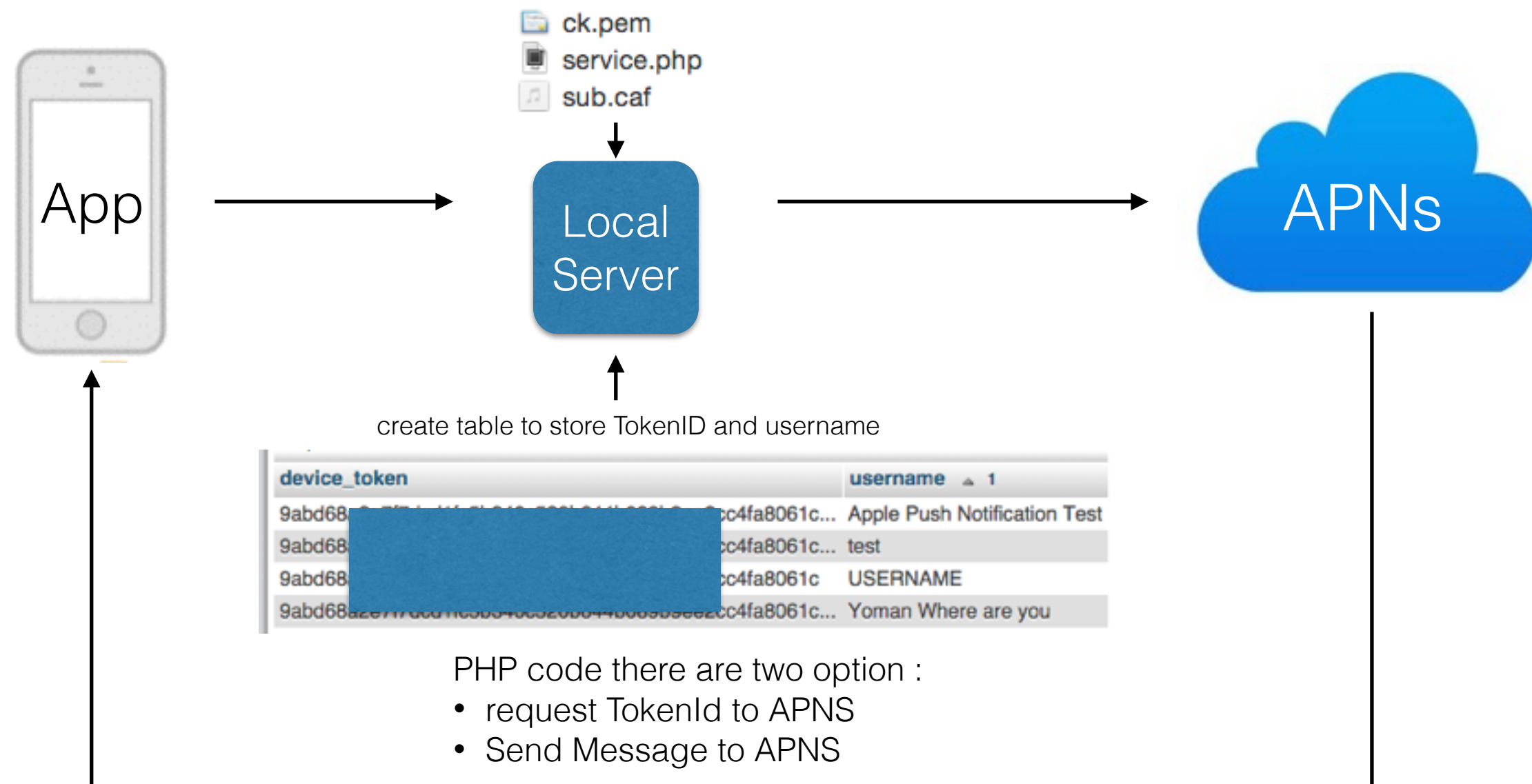
that alert just support version below IOS 8.0
go and find code that support > IOS 8.0

AppDelegate.m

```
- (void)application:(UIApplication*)application didRegisterForRemoteNotificationsWithDeviceToken:(NSData*)deviceToken{
    NSLog(@"My token is: %@", deviceToken);
}

- (void)application:(UIApplication*)application didFailToRegisterForRemoteNotificationsWithError:(NSError*)error{
    NSLog(@"Failed to get token, error: %@", error);
}
```

5. Process of PHP code



<https://www.dropbox.com/sh/nzy2f5oru5lpiq4/AAAAa1FpZhK65KCoEMrPvkozta?dl=0>

Ex

Resgister DeviceID and UserName to Server:

--> `http://localhost/PushTesting/service.php?token=DEVICE_TOKEN&cmd=reg&name=USERNAME`

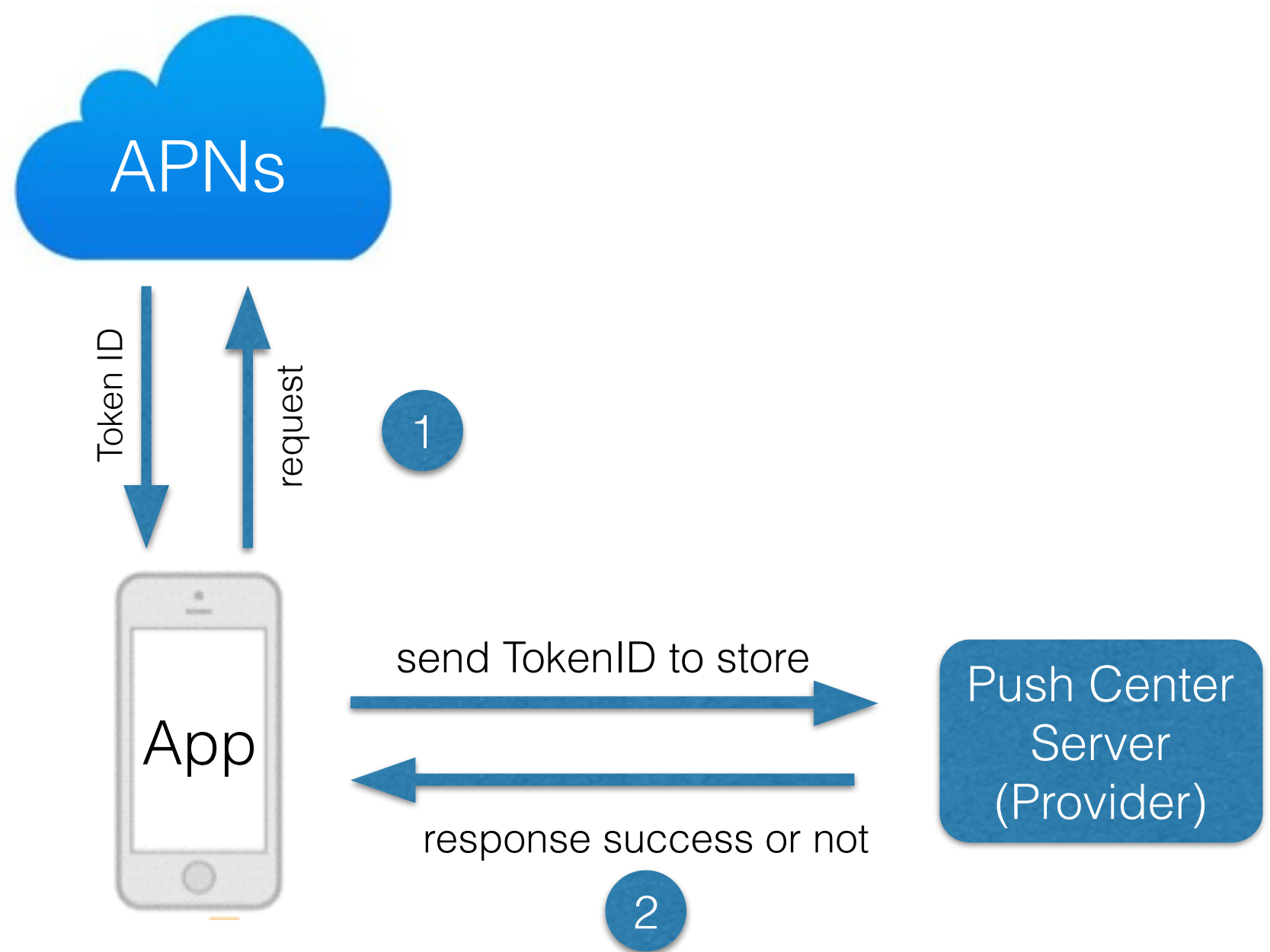
Send message to DeviceID :

--> `http://localhost/PushTesting/service.php?token=DEVICE_TOKEN&cmd=msg&msg=YOURMESSAGE`

so just request that URL from mobile and push msg will work

6. Process of APNs

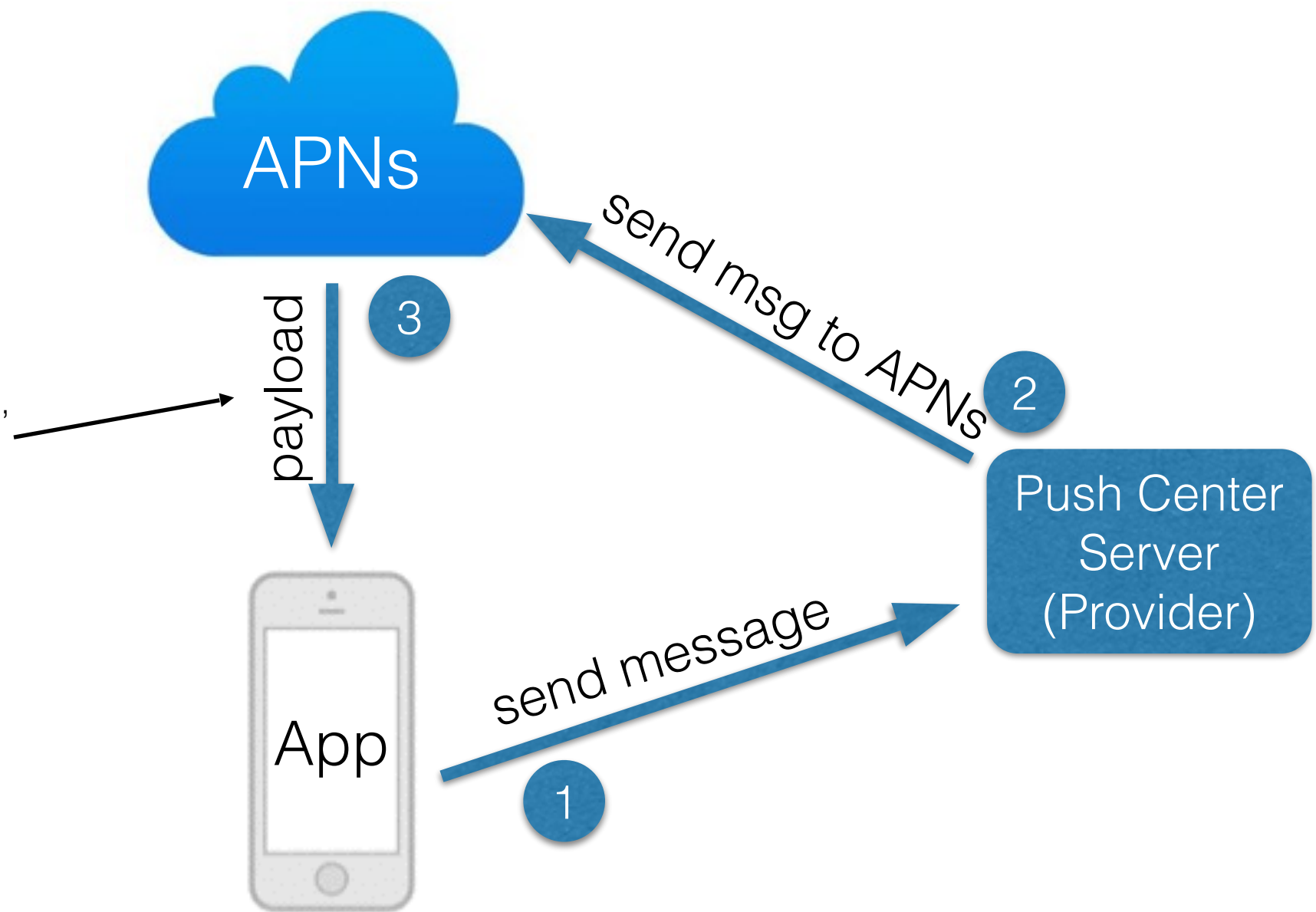
step 1



step 2

JSON

```
{
  "aps": {
    "alert": {
      "body": "Yoman is here",
    },
    "badge": 1,
  },
}
```



7. Example

-after got TokenID send it to server

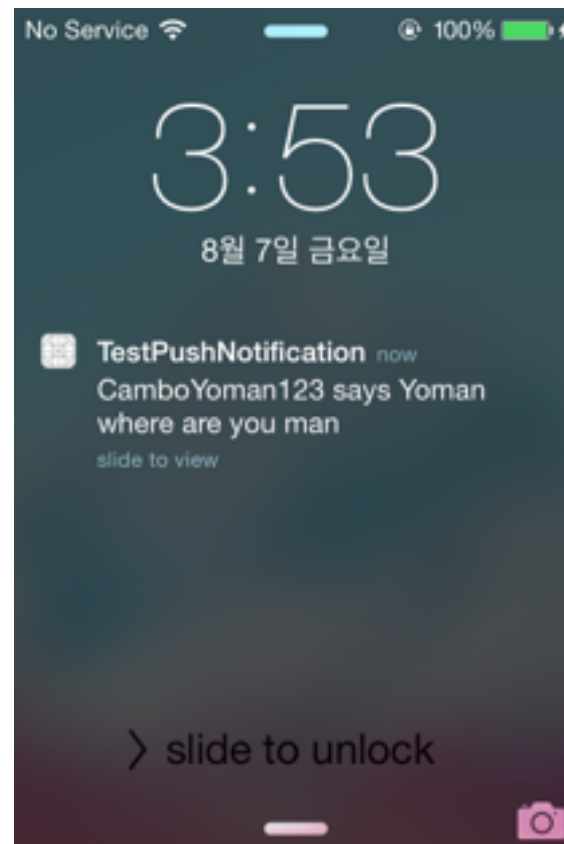
create table to store TokenID and username

device_token	username
9abd68...cc4fa8061c...	Apple Push Notification Test
9abd68...cc4fa8061c...	test
9abd68...cc4fa8061c...	USERNAME
9abd68...cc4fa8061c...	Yoman Where are you

use this for send : http://localhost/PushTesting/service.php?token=DEVICE_TOKEN&cmd=reg&name=USERNAME

-after send msg to APNs by device_token in DB

use this to send : http://localhost/PushTesting/service.php?token=DEVICE_TOKEN&cmd=msg&msg=YOURMESSAGE



Note : if you want to send to and another user or device . you have to make sql to compare. like select device_token from tableA where username = 'yoman'. and send with the result device_token and with message text to APNs provider

Note : .PHP if cmd = reg is just for register to server

.PHP if cmd = msg is just select all device_token and send to APNs provider

