

How to compile manually iOS apps?

The Certificates

An iOS publication under one developer account needs 4 files, one file that will sign the certificates and 3 certificates. The file that will sign the certificates and one of the 3 certificates are used for all apps published under one developer account, and the two other certificates must be created each time you will publish a new app.

The ones that are used for all apps published under one developer account:

These 2 files will be created once (the first time you are going to publish under this developer account). It is the CertSigningRequest (CSR) and the iOS distribution certificate (iOS_Distribution). The CSR is created from your computer (must be a Mac) from the Keychain Access, and the iOS_Distribution is created from the Member Center at developer.apple.com.

These two items must be stored in the main certificates folder for this developer account on your computer (it will be better in term of organization if you are willing to publish other apps under this account).

Thus if we consider this example, for the client named "John Doe" who has his own developer account, on your computer there will be these folders with this organization:

- IOS apps
 - John Doe
 - Certificates → CSR and iOS_Distribution for John Doe account must be stored here after you have created them
 - John Doe 1st app
 - Certificates
 - Screenshots

You will use this CSR and iOS_Distribution certificate for all the apps that you will publish for John Doe, if there are several apps to publish under this account. You create once and you use them each time you publish an app for John Doe under his account. For your information, the Keychain Access to create the CSR is in your Mac in Applications > Utilities.

The two other certificates are the Apple Push Services certificate and the Mobile Provisioning Profile. You will have to create them for each app, as you can see in the video. You will stored them in the certificates folder which is in the app folder. Thus if we consider the previous example, for the client named "John Doe" who has his own developer account:

- IOS apps

- John Doe

- Certificates

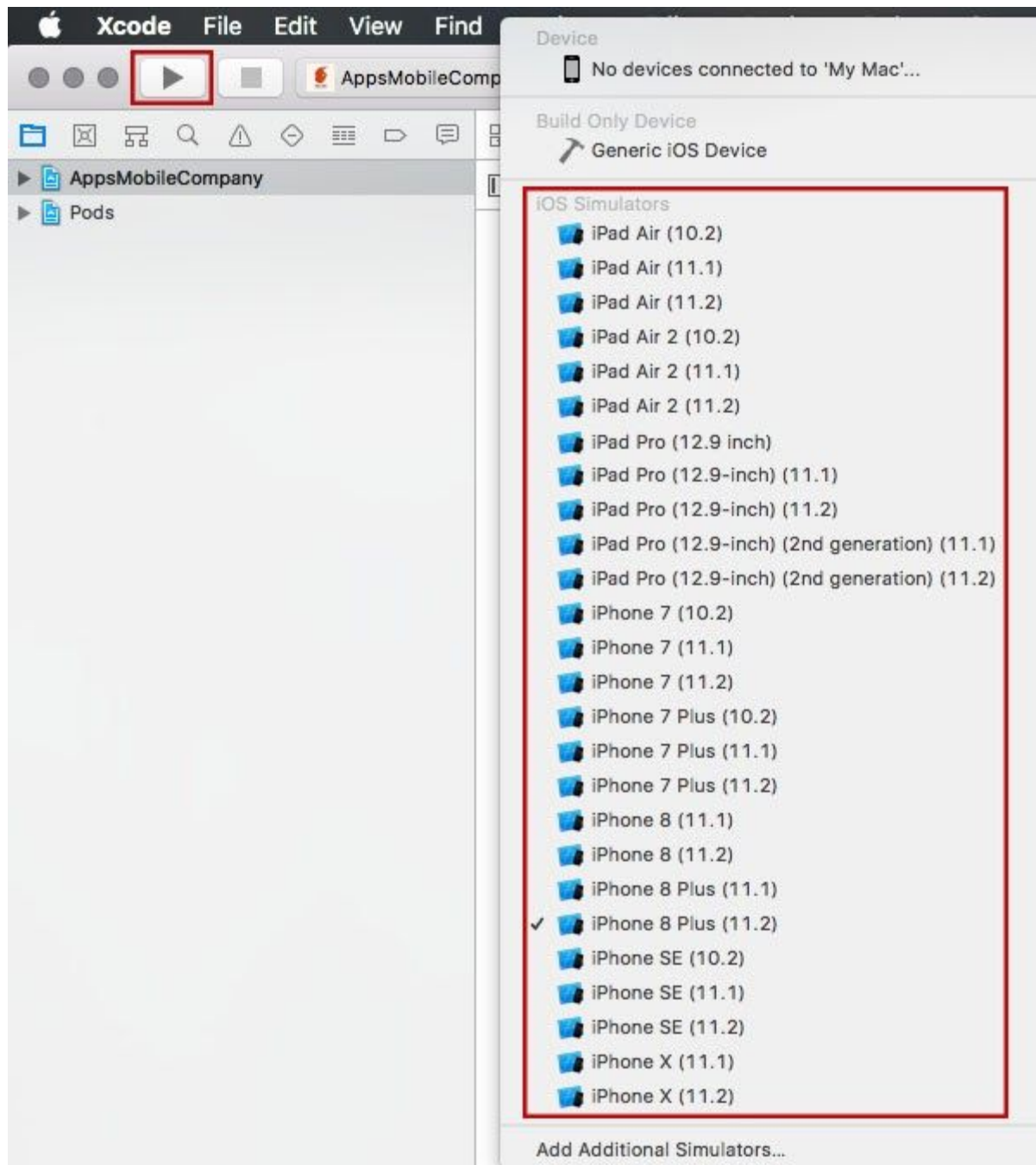
- John Doe 1st app

- Certificates → Apple Push Services certificate and Mobile Provisioning Profile for John Doe 1st app must be stored here after you have created them

- Screenshots

To create the screenshots

You will need some screenshots to fill out the app sheet on iTunes. You will be able to make them with Xcode and especially with the iOS Simulator delivered with Xcode. Open the Xcode project that is in the ios_source folder (see in the video when opening the project with Xcode), then choose the target, iPhone 8 Plus for instance, and click on the Play button (For your info, the ios_sources of an app can be downloaded from its page from your backoffice > Manage > Applications):



Take some screenshots of the app for this target using CMD+S from your keyboard, and do it for each target you want screenshots. Before switching from a target to another don't forget to click on the "stop" button in Xcode.

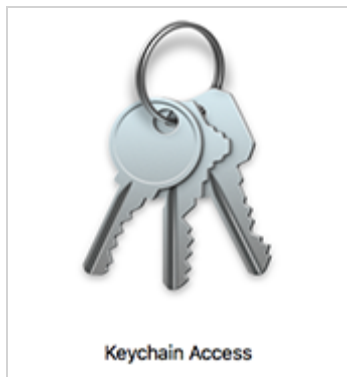
Step to compile an iOS app

Create the items related to the developer account with which you are going to publish

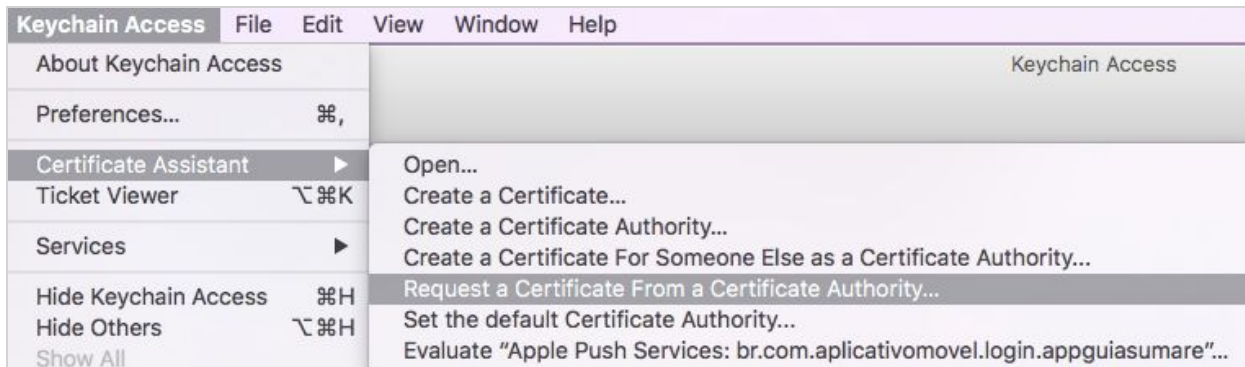
You have to create a CSR file and an iOS Distribution certificate. And you will be able to use these items for all of the next apps you will have to publish under this developer account → keep them preciously once created.

The CSR File

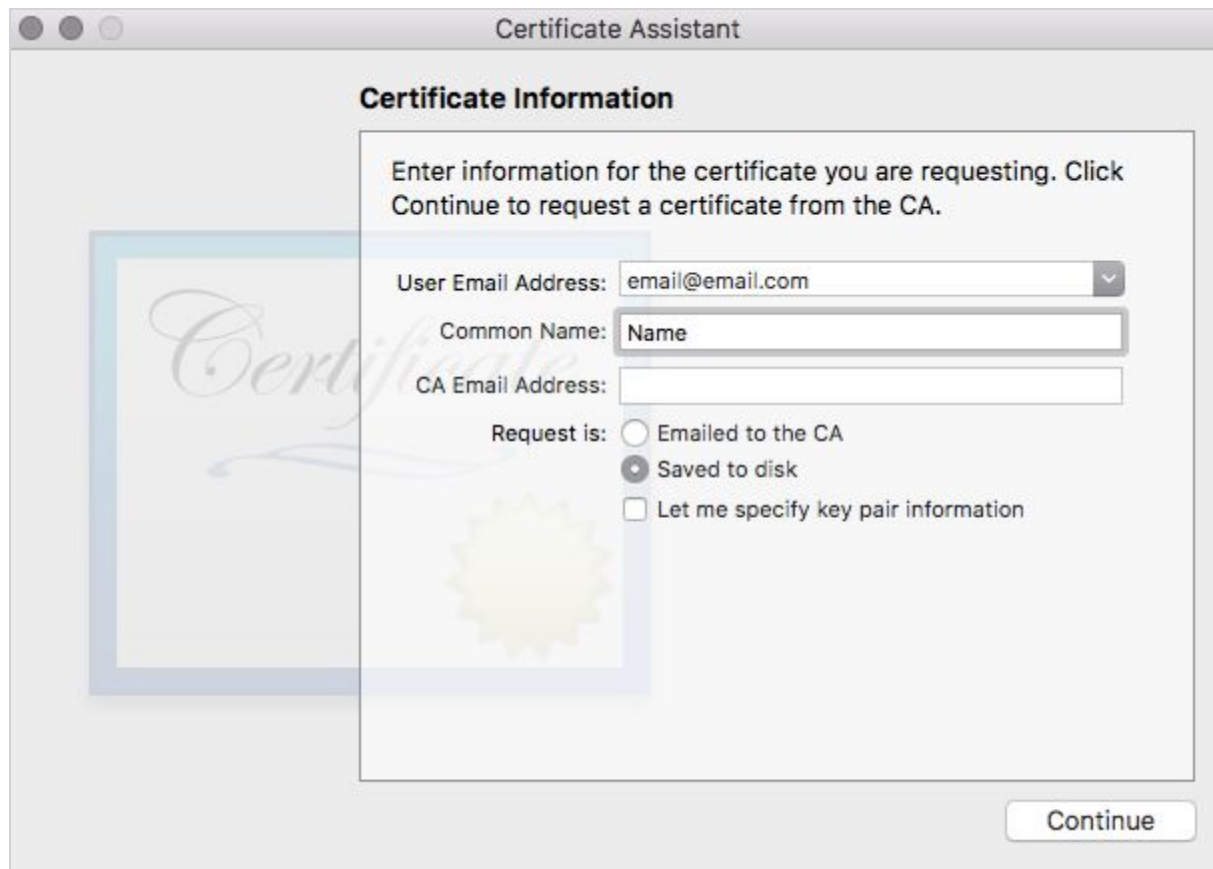
It must be created from the keychain access tool of your Mac. Open it from Applications > Utilities > Keychain Access:



From the menu select “Keychain access” > “Certificate Assistant” > “Request a Certificate From a Certificate Authority”:



In the window that opens, enter the email of the developer account (or any other email address, it doesn't have any impact), and enter the name of your developer account, and select “Saved to disk”:



Save this CSR to your disk, in a folder related to the certificates of your developer account.

Close the window.

The iOS Distribution certificate


– Connect at <https://developer.apple.com/membercenter/index.action> with the developer account

– Click on “Certificates, Identifiers & Profiles”



Certificates, Identifiers & Profiles

Manage the certificates, identifiers, profiles, and devices you need to develop and distribute apps.

– Click on “Certificates” in the menu on the left, then click on  :

Certificates, Identifiers & Profiles

Certificates

Certificates 

– Choose “iOS Distribution (App Store and Ad Hoc)” and click on “Continue”:

iOS Distribution (App Store and Ad Hoc)

Sign your iOS app for submission to the App Store or for Ad Hoc distribution.

– Sign your iOS Distribution certificate with the CSR file you have created before.

– Download the iOS Distribution certificate that has been created.

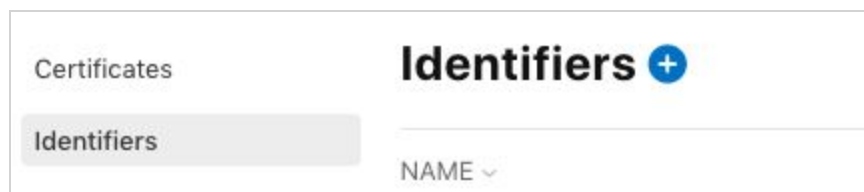
– Double click on the iOS Distribution certificate (ios_distribution.cer) to add it to the Keychain Access.

Create the items related to the app you are going to publish

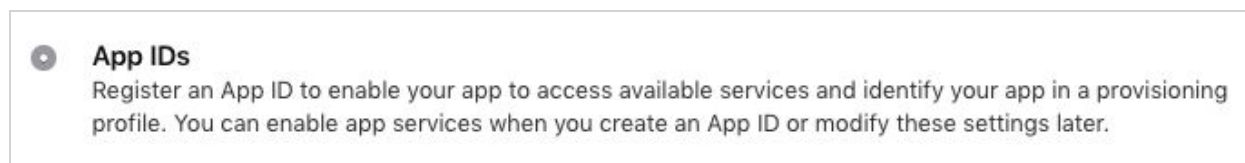
The Apple Push Services certificate

Thanks to this certificate you will be able to add push notifications, in-app purchase, and other services to your app. In our case, only the push notification service is interesting.

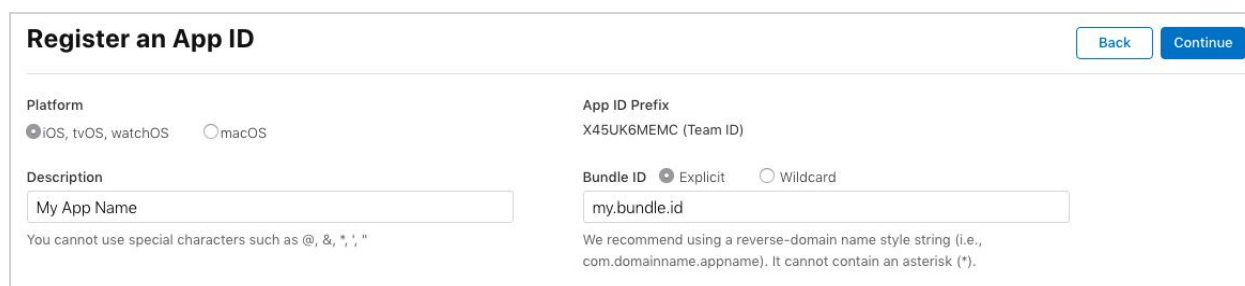
– Click on “Identifiers” in the menu on the left, then click on  :



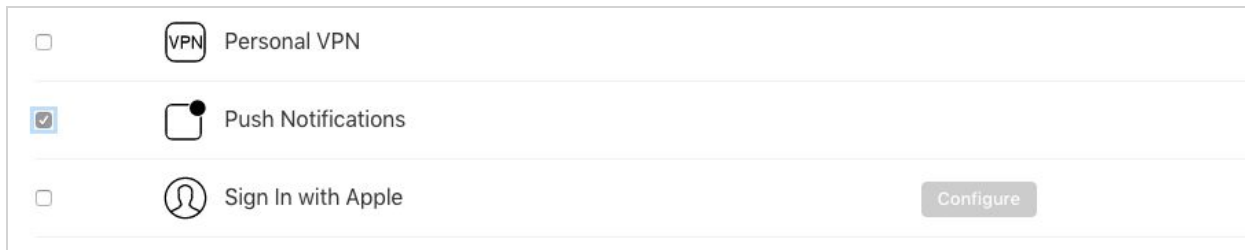
– Choose “App IDs” and click on “Continue”:



– Enter your app name, the Bundle ID of the app:

A screenshot of the 'Register an App ID' form in the Apple Developer portal. The form has a title 'Register an App ID' and two buttons: 'Back' and 'Continue'. The form is divided into two columns. The left column has a 'Platform' section with radio buttons for 'iOS, tvOS, watchOS' (selected) and 'macOS'. Below that is a 'Description' section with a text input field containing 'My App Name' and a note: 'You cannot use special characters such as @, &, *, ', ;, *'. The right column has an 'App ID Prefix' section with the value 'X45UK6MEMC (Team ID)'. Below that is a 'Bundle ID' section with radio buttons for 'Explicit' (selected) and 'Wildcard', and a text input field containing 'my.bundle.id'. A note below the Bundle ID field reads: 'We recommend using a reverse-domain name style string (i.e., com.domainname.appname). It cannot contain an asterisk (*).'

– Go below and select “Push Notifications” in the list of “Capabilities“:

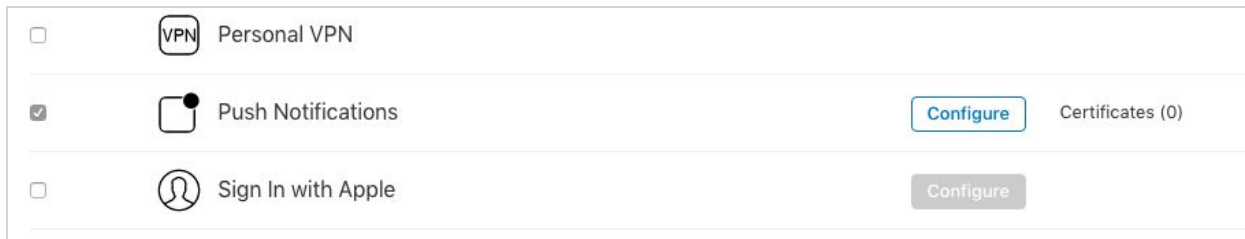


– Click on “Continue”.

– Then click on “Register”.

– In the list, find the App ID you have just created and click on it.

– Go at the bottom and for Push Notifications, click on “Configure”:



– In “Production SSL Certificate” section click on “Create Certificate”:

Apple Push Notification service SSL Certificates

To configure push notifications for this App ID, a Client SSL Certificate that allows your notification server to connect to the Apple Push Notification Service is required. Each App ID requires its own Client SSL Certificate. Manage and generate your certificates below.

Development SSL Certificate

Create an additional certificate to use for this App ID.

Create Certificate

Production SSL Certificate

Create an additional certificate to use for this App ID.

Create Certificate

Done

– Click on “Choose File” and upload the CSR file you have created before for this developer account (the same CSR you have used to create the iOS Distribution certificate for this developer account):

Create a New Certificate

Back

Continue

Certificate Name

Apple Push Notification service SSL (Sandbox & Production)

Upload a Certificate Signing Request

To manually generate a Certificate, you need a Certificate Signing Request (CSR) file from your Mac.

[Learn more >](#)

Choose File

CertificateSigningRequest.certSigningRequest

– Download the Apple Push Services certificate (aps.cer) that has just been created. Store this certificate in a folder related to this app on your Mac.

Download Your Certificate Revoke Download


Certificate Details



Certificate Name my.bundle.id.app	Certificate Type Apple Push Services	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.
Expiration Date 2020/07/05	Created By Johnny Cage	

– Click on “< All Certificates“.

-Double click on the Apple Push Services certificate (aps.cer) to add it to the Keychain Access.

The Provisioning Profile

– Click on “Profiles” in the menu on the left, then click on  :

Certificates	Profiles 
Identifiers	NAME 
Devices	1984 - Distribution
Profiles	2099 App - Distribution
Keys	

– Choose “App Store” under “Distribution“:

Distribution

- Ad Hoc**
Create a distribution provisioning profile to install your app on a limited number of registered devices.
- tvOS Ad Hoc**
Create a distribution provisioning profile to install your app on a limited number of registered tvOS devices.
- App Store**
Create a distribution provisioning profile to submit your app to the App Store.
- tvOS App Store**
Create a distribution provisioning profile to submit your tvOS app to the App Store.
- Mac App Store**
Create a distribution provisioning profile to submit your app to the Mac App Store.

– Select the App ID of your app, and click on “Continue”:

Generate a Provisioning Profile

[Back](#) [Continue](#)

Select Type > Configure > Generate > Download

Select an App ID
If you plan to use services such as Game Center, In-App Purchase, and Push Notifications, or want a Bundle ID unique to a single app, use an explicit App ID. If you want to create one provisioning profile for multiple apps or don't need a specific Bundle ID, select a wildcard App ID. Wildcard App IDs use an asterisk (*) as the last digit in the Bundle ID field.

App ID:
The App (X45UK6MEMC.com.siberiancms.ios5b6811ebcfa6d) ▾

– Select the iOS Distribution certificate associated to the CSR you have used to sign the Apple Push Services Certificate you have created just before (it is very important to use the iOS Distribution which has been created with the CSR used to sign the aps.cer of this app. Don't forget it, and don't forget that in case you create a new aps.cer for this app with another CSR you will have to create a new provisioning profile too with the iOS Distribution certificate made with this CSR, and then to publish an update of your app on the App Store)

Generate a Provisioning Profile Back Continue

Select Type > Configure > Generate > Download

Select Certificates
Select the certificates you wish to include in this provisioning profile. To use this profile to install an app, the certificate the app was signed with must be included.

- Xtraball (iOS Distribution) Mar 10, 2020
- Xtraball (iOS Distribution) Mar 10, 2020
- Xtraball (iOS Distribution) Jun 05, 2020

– Click on “Continue”.

– Name the Provisioning Profile and click on “Continue”

Generate a Provisioning Profile Back Generate

Select Type > Configure > Generate > Download

Review, Name and Generate.
The name you provide will be used to identify the profile in the portal.

Provisioning Profile Name

Type
iOS Distribution

App ID
The App(X45UK6MEMC.com.siberiancms.ios5b6811ebcfa6d)

Certificates
1 Selected

– Download the Provisioning Profile that has just been created. Store this certificate in the folder related to this app on your Mac, and which contains the Apple Push Services certificate created just before.

Generate a Provisioning Profile Download

Download and Install
Download and double click the following file to install your Provisioning Profile.

Profile Name
My Siberian App - Distribution

Profile Type
iOS Distribution

App ID
X45UK6MEMC.com.siberiancms.ios5b6811ebcfa6d

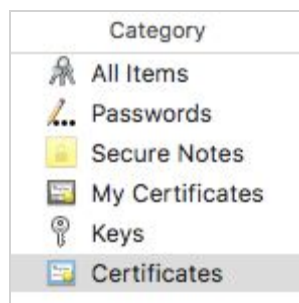
Expires:
2020/06/05

Compiling the app

- Download your iOS source code
- Unzip the folder and open the .xcworkspace file by double-clicking it
- **Follow given video video:**

Creating the .pem Certificate

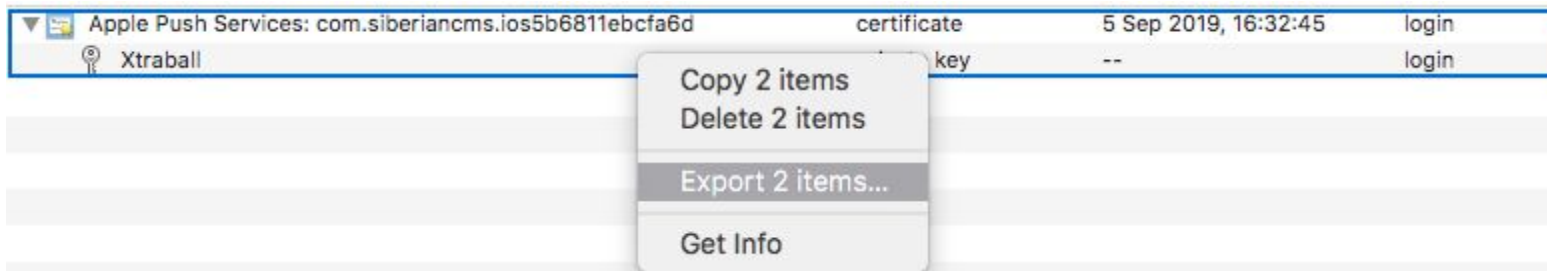
- Go in the Keychain Access on your Mac
- From the left menu, select “Certificates”



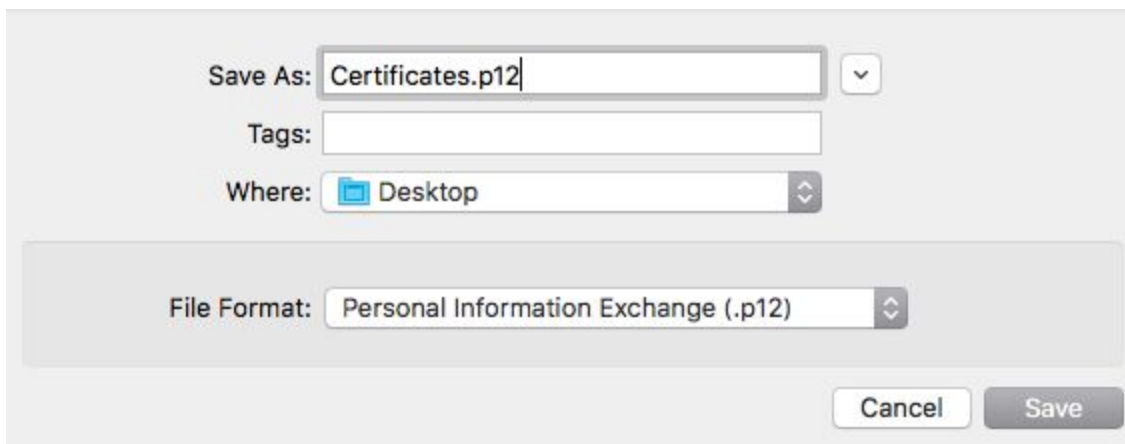
- In the search bar at the upper right, enter the bundle ID of your app
- Click on the arrow on the left of your “Apple Push Services” certificate, a private key must be attached under it

Apple Push Services: com.siberiancms.ios5b6811ebcfa6d	certificate	5 Sep 2019, 16:32:45	login
Xtraball	private key	--	login

- Select the private key and the certificate
- Right-click on them
- Select “Export 2 items...”



– Choose your desktop as the destination (it can be another location but you will have to adjust the code we will give you after)



– Choose a password of your own



– Open the Terminal console of your Mac (From Applications > Utilities > Terminal)

– In the Terminal, be sure to be located on your desktop (you must have “Desktop” displayed and then your username)

```
Mac-8:~ admin$ cd Desktop/  
Mac-8:Desktop admin$
```

– Enter this command:

“openssl pkcs12 -in Certificates.p12 -out certificat.pem -nodes -clcerts”

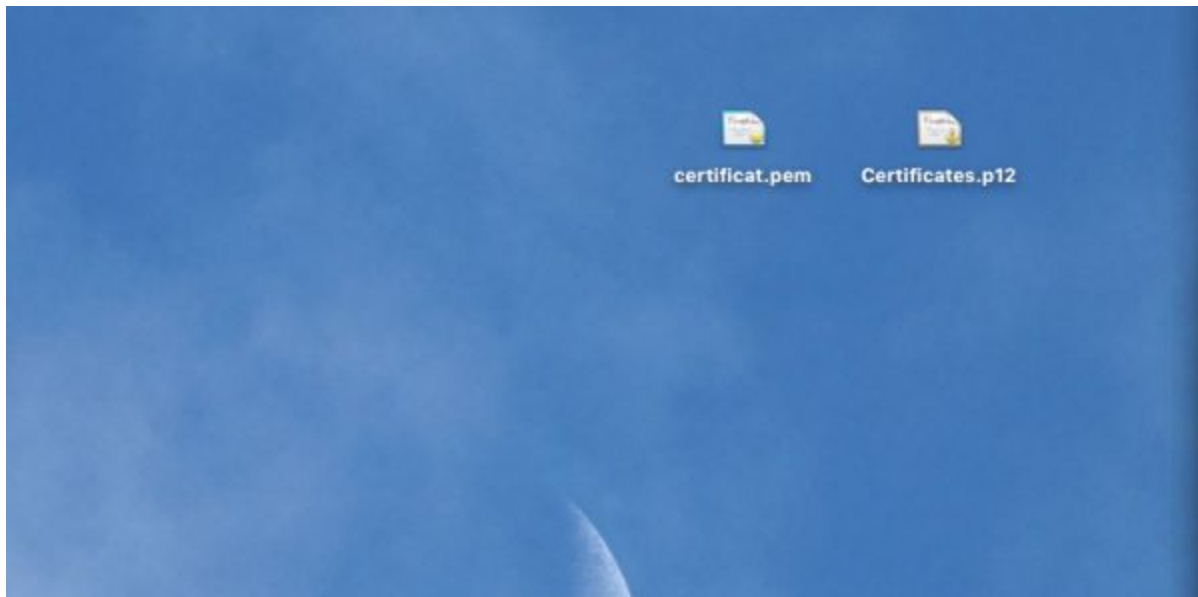
(If the terminal answers that there is no file named “Certificates.p12”, go on your desktop and check what is the name of this certificate .p12, and enter this name rather than Certificates.p12 in the command line)

```
Mac-8:Desktop admin$ openssl pkcs12 -in Certificates.p12 -out certificat.pem -nodes -clcerts  
Enter Import Password:
```

– Enter the password you have used just before to export your certificate on your desktop

```
Enter Import Password:  
MAC verified OK  
Mac-8:Desktop admin$
```

– On your desktop you have now two certificates: a Certificates.p12 and a certificat.pem.



– Move these files to the folder of your app on your Mac (with the provisioning profile and the aps.cer)

Today

-  aps.cer
-  certificat.pem
-  Certificates.p12
-  My_App_Di...ileprovision

– Upload this certificat.pem on the app sheet in your backoffice (Pls send this certificate to app builder support team to upload)

– You're done!