Steps to Renew APNS Certificate for use with PushSharp Library:  
  
Get with Brian to determine if this is still be best process or if I should be following the steps at:  
<https://github.com/Redth/PushSharp/wiki/How-to-Configure-&-Send-Apple-Push-Notifications-using-PushSharp>

Root Folder = coderepo/\_misc/Certificates & Keys/iOS

1. Log in to iOS Developer Center. Use the TeleMed team.
2. Go to the list of App Ids. Expand MyTeleMed and click the Edit button. Assuming the Production SSL Certificate is about to expire, there should be an option to create another certificate.
3. Create the certificate. If possible use the existing .certSigningRequest file located at Root Folder/Cert Signing Requests/Production/MyTeleMed APNS Production.certSigningRequest
4. Download the generated certificate to coderepo Root Folder/Certificates (CER)/Production/MyTeleMed APNS Production.cer
5. Open the .cer file above in OSX Keychain.
6. Under My Certificates, the Apple Production IOS Push Services: com.solutionbuilt.telemed should exist with the new expiration date. Expand it. Click to highlight both the certificate and its private key, then right-click and Export 2 items... Save the p12 file to Root Folder/P12 Files/Production/MyTeleMed APNS Production Private Key.p12 (don't create a password)
7. Using command line, convert Identity Certificate from Step 4 to PEM file:  
   openssl x509 -in MyTeleMed\ Production\ APNS\.cer -inform DER -out MyTeleMed\ Production\ APNS\ Identity\ Certificate.pem -outform PEM
8. Using command line, convert Private Key P12 File from Step 6 to PEM file (use password: telemed4321$$):  
   openssl pkcs12 -nocerts -in MyTeleMed\ Production\ APNS\ Private\ Key.p12 -out MyTeleMed\ Production\ APNS\ Private\ Key.pem
9. Move both PEM files into the Root Folder/PEM Files/Production folder.
10. Test the PEM files against Apple's APNS server:  
    openssl s\_client -connect gateway.push.apple.com:2195 -cert MyTeleMed\ Production\ APNS\ Identity\ Certificate.pem -key MyTeleMed\ Production\ APNS\ Private\ Key.pem -debug -showcerts -CAfile ../entrust\_2048\_ca.pem
11. Combine both PEM files into a P12 File (use password: telemed4321$$):  
    openssl pkcs12 -export -in MyTeleMed\ Production\ APNS\ Identity\ Certificate.pem -out MyTeleMed\ Production\ APNS.p12 -inkey MyTeleMed\ Production\ APNS\ Private\ Key.pem
12. Send MyTeleMed APNS Production.p12 to Telemed.