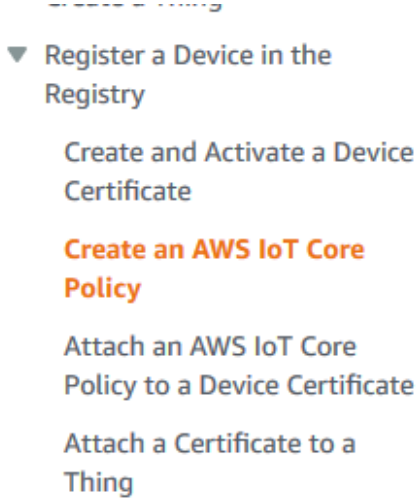
1 AWS IoT Core Client

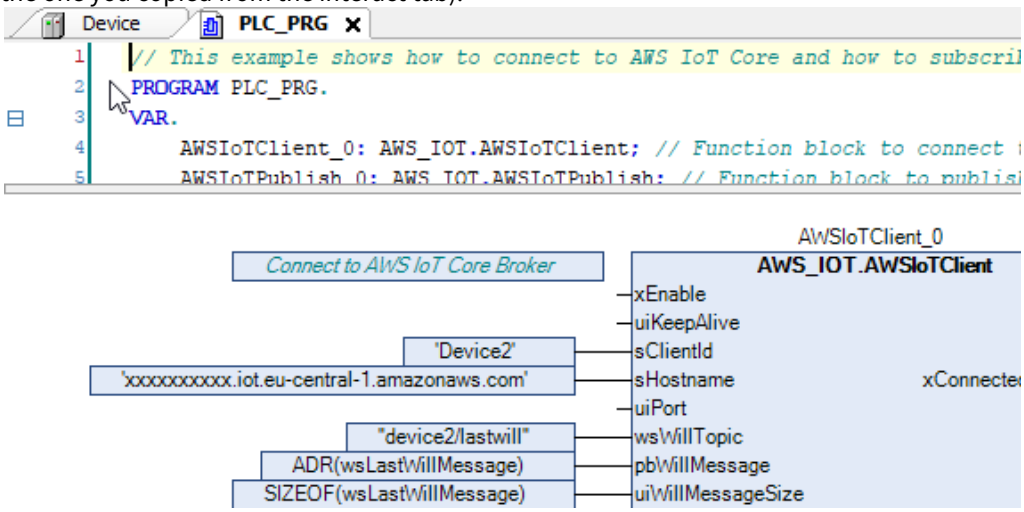
1. Install “Security Agent” addon form store.codesys.com[1]
2. Choose a binary to install from https://wiki.openssl.org/index.php/Binaries (first one worked for me)
3. Go to aws.amazon.com[2] and sign up
4. Read all the docs with Amazon! We follow slightly different steps.



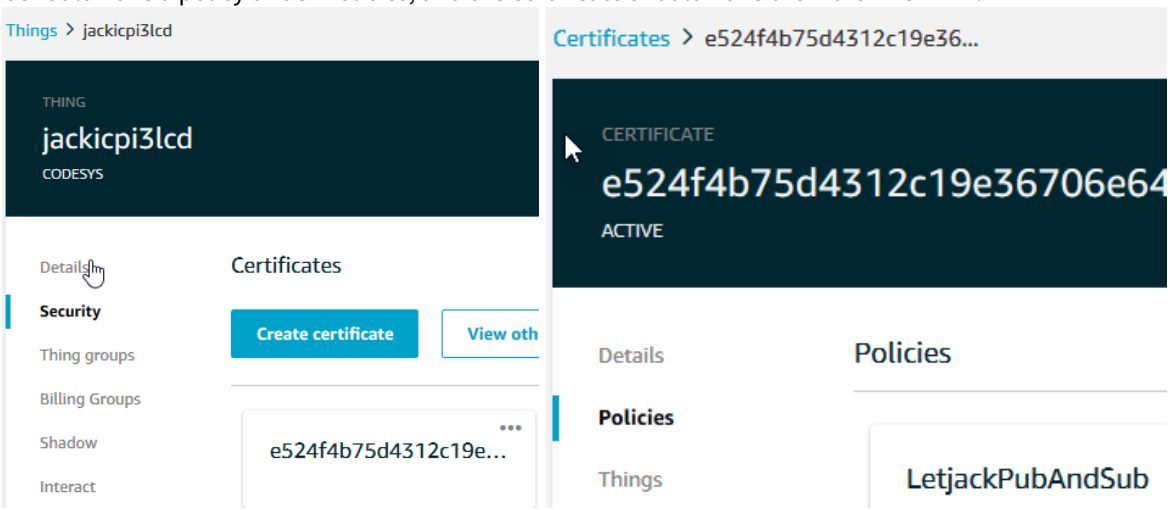
5.
6. Click “All Services” and IOT Core
7. Click “Manage”
8. Click “Register a Thing”
9. Click “Create a single Thing”
10. Name him “jackicpi3lcd”
11. Create a type “CODESYS” / “Things connected with CODESYS”
12. Don’t bother with group, skip all the optional things and hit “next”
13. Create thing without Certificate
14. Hit “Done”
15. Open your THING and go to “Interact”
16. Copy the rest API endpoint under HTTPS, eg. “xxxxxxxxxxxxx-ats.iot.us-east-2.amazonaws.com[3]”
17. Back to the main menu of AWS IoT, Under Menu “Secure > Policies”, Press Create new policy. Note that your account ID is different from the URL in the Rest API mentioned above! It is given as the default Resource when you create a policy.
18. Name it LetJackPubAndSub
19. You want to be able to connect with a client ID, publish the topic and the last will, Subscribe t othe topic, and then once subscribed, you also want ot receive messages from the topic.
20. Action = iot:Connect
21. Resource ARN = arn:aws:iot:<AWS Region>:<AccountID, not same as Rest API URL>:client/jackicpi3lcd
22. Effect = Allow
23. Click Add Statement
24. Action = iot:Publish
25. Resource ARN = arn:aws:iot:<AWS Region>:<AccountID, not same as Rest API URL>:topic/hello/aws, arn:aws:iot:<AWS Region>:<AccountID, not same as Rest API URL>:topic/jackicpi3lcd/lastwill
26. Effect = Allow
27. Click Add Statement
28. Action = iot:Subscribe

1 http://store.codesys.com
2 http://aws.amazon.com
3 http://xxxxxxxxxxxxx-ats.iot.us-east-2.amazonaws.com

29. Resource ARN = arn:aws:iot:<AWS Region>:<AccountID, not same as Rest API URL>:topicfilter/hello/aws
30. Effect = Allow
31. Click Add Statement
32. Action = iot.Receive
33. Resource ARN = arn:aws:iot:<AWS Region>:<AccountID, not same as Rest API URL>:topic/hello/aws
34. Effect = Allow
35. Click Create
36. 

37. Open the AWS_IoT_Core_Client_Example.project in CODESYS
38. Right click the Device in Device Tree, select "Update Device...", then select your target PLC, before hitting OK (Important step for every example project!)
39. Under the Application called AWSPubSub, Open the PLC_PRG and change these settings in the screenshot below to match your THING (Change the two instances of ‘Device2’ to jackicpi3lcd, Change the Hostname to the one you copied from the interact tab).
40. 

41. Download and run
42. Open the visualisation and hit xEnable for the AWSIoTClient. It will NOT show that it is xConnectedToBroker, because we haven't configured the permissions yet.
43. Open the PLC shell (Device -> PLC-Shell).
44. Specify cert-getapplist. -> A component with the specified device name and a number is displayed. You want the one that says “jackicpi3lcd”

45. Specify cert-createcsr <number> and use the number from the previous step. The creation of the CSR file can take several seconds. A corresponding message is displayed in the device log (Device -> Log) after it has been created.
46. Open (Device -> Files) and copy the CSR file from the cert/export directory to the local file system.
47. Open a command prompt and type something like: openssl.exe req -in "D:/prj/WIP/AWS IOT CORE/6_jackicpi3lcd.csr" -inform der -out "D:/prj/WIP/AWS IOT CORE/jackicpi3lcd.csr"
48. go back to the AWS management console
49. open your thing and go to the security console
50. hit “View other options”
51. hit “create with csr”
52. use the output from the above openssl command (“jackicpi3lcd.csr”)
53. Hit “Upload File”. If it doesn’t say successful after 10 seconds, you’ve grabbed the wrong file.
54. Download the certificate it gives you (depending on your browser, will actually save as .txt for some reason)
55. Hit the link to download a root certificate. Grab Amazon root ca1, Root CA3, and startfield Root CA Certificates.
56. Press Activate.
57. Press Attach a policy
58. Select the policy you created earlier
59. Now it won’t have added the certificate to the thing for some reason.. Go back to the main menu and select Secure > Certificates, select the new one you made.
60. Actions > Activate
61. Actions > Attach Thing > jackicpi3lcd > Attach
62. So now if you go to Manage > Thing > jackicpi3lcd, it should have a certificate under security. If you click that it should have a policy under Policies, and the certificate should have the word “ACTIVE”.
63. 

64. Back to codesys now.
65. Security screen > Devices > Hit Refresh > Click on “Own Certificates”
66. Press the button just to the left of Owned certificates that looks like a PLC with a green downwards arrow.
67. Change the file filter to all files, change the extension of the file you downloaded from .txt to .crt, then select it and hit open.
68. Go to trusted certificates folder now, and install all three root certificates (same button) from Amazon, again changing the filter if needed.
69. Cold reset the controller to take the new certificates.
70. Open the visualization, turn on xEnable again for AWSIoTClient. You should get a green “connected to broker” lamp.
71. xEnable the AWSIoTSubscribe
72. Enable the AWSIoTPublish

73. Getting values, everything is good! (The topic and payload you send via Publish should be seen in the PLC_PRG.AWSIoTSubscribe_0)