Internet of Things (IoT) – MQTT with Amazon Web Services (AWS)

For Use with the JNIOR Series 4

Last updated: December 4, 2019

The following information describes how to use the JNIOR Series 4 and its MQTT communication protocol with Amazon Web Services.

If you have any questions or want to use the JNIOR and MQTT with another broker, please contact INTEG via e-mail at **support@integpg.com** or via phone at 724-933-9350 with any questions. INTEG can adapt the MQTT application running on the JNIOR to meet your specific needs.

Overview

The INTEG JNIOR automation controller is capable of being an edge device for the Internet of Things (IoT) applications using the MQTT protocol. The JNIOR can both publish and subscribe to topics using an MQTT broker.

The JNIOR implements the complete MQTT protocol including CONNECT, CONNACK, PUBLISH, PUBACK, SUBSCRIBE, SUBACK and UNSUBSCRIBE. The JNIOR can also provide all three Quality of Service levels.

The JNIOR topics are structured as follows:

DIGITAL INPUTS

jnior/'serial number'/DIN#

where DIN# is 1 – 12 representing the JNIOR digital inputs

and the data provided in the payload is as follows:

"State" - "HIGH" or "LOW" (on or off)
"Counter" - value of the counter (counter increments each time input goes low to high)
"UsageMeter" - value of the usage meter (timer increases when the input is 'high')

RELAY OUTPUTS

jnior/'serial number'/ROUT#

where ROUT# is 1 – 16 representing the JNIOR relay outputs

and the data provided in the payload is as follows:

"State" - "HIGH" or "LOW" (on or off)
"UsageMeter" - value of the usage meter (timer increases when the output is 'high')

ANALOG INPUTS

jnior/'serial number'/AI#

where AI# is 1 – 8 representing the JNIOR analog inputs

and the data provided in the payload is as follows:

"value" - analog input reading

ANALOG OUTPUTS

jnior/'serial number'/AO#

where AI# is 1 – 4 representing the JNIOR analog outputs

and the data provided in the payload is as follows:

"value" - analog output reading

TEMPERATURE SENSORS

jnior/'serial number'/temperature

where temperature is a digital temperature sensor connected to the JNIOR

and the data provided in the payload is as follows:

"tempF" - temperature in Fahrenheit
"tempC" - temperature in Celsius

HUMIDITY SENSORS

jnior/'serial number'/humidity

where humidity is a digital humidity sensor connected to the JNIOR

and the data provided in the payload is as follows:

"value" - humidity reading

ALL DATA

jnior/'serial number'/# acts as a wildcard and the broker can subscribe to all topics

In all of the above, 'serial number' is the serial number of the JNIOR which distinguishes the source of each topic.

The MQTT application when installed on the JNIOR enables the JNIOR to both publish and subscribe to topics. The application can be installed on any of the JNIOR Series 4 controllers – Models 410, 412, 414. All of the JNIOR digital inputs, relay outputs, and JNIOR expansion modules such as analog signals, temperature signals and humidity signals can be integrated. The digital signal status is only published when the status is changed. The analog signals can be sent on a change in value or time basis.

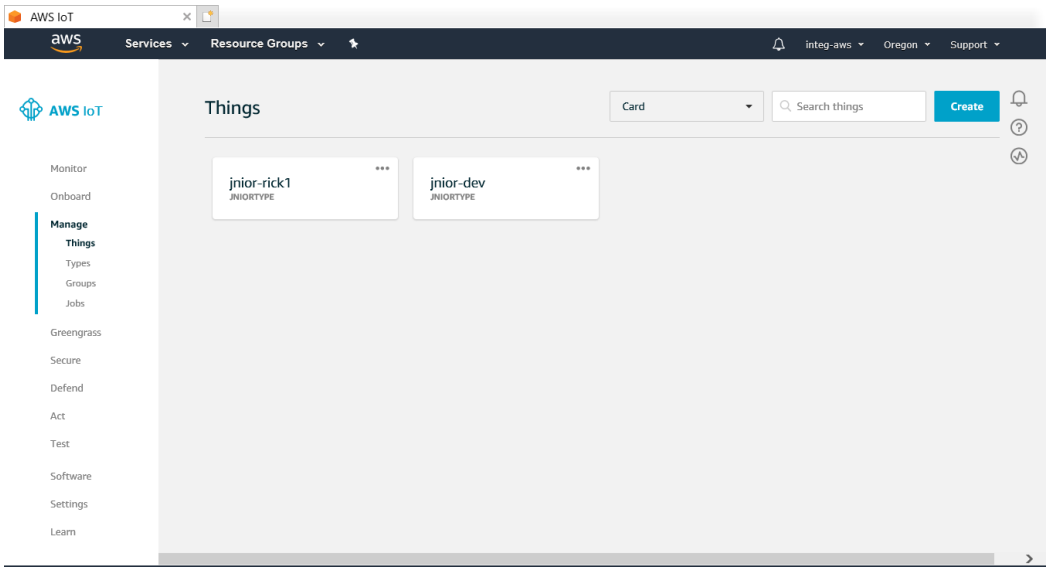
Amazon Web Services

The JNIOR is capable of interacting with Amazon Web Services (AWS) using a secure MQTT protocol connection. AWS brings a variety of features and functions for gathering data from IoT devices and then acting upon this data whether it is through data storage, text alerts, email messages, etc. How to use the 'data' is very dependent upon the user's application for the data.

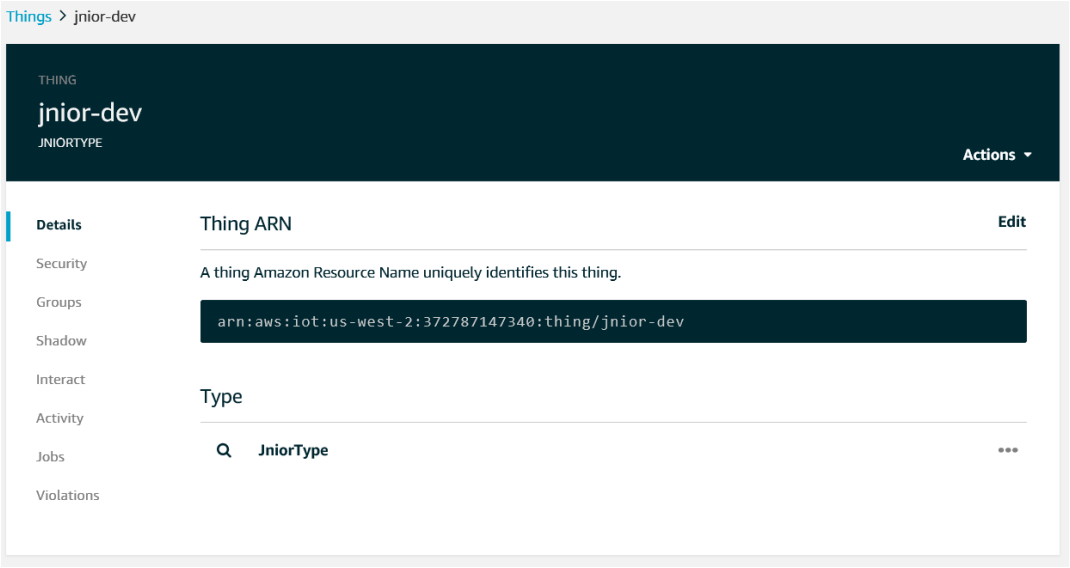
The remainder of this document will describe how to enable the JNIOR to communicate with AWS and discuss a simple example where a text or email via AWS is sent each time a JNIOR input goes 'on' or a temperature exceeds a predefined limit. The document does not describe all the various features of AWS nor go into detail on how to exactly configure and use AWS.

To have the JNIOR connect to AWS, you need to create an AWS 'Thing'. Since AWS uses a secure connection, the first step is to create the necessary security certificates on the AWS site and then transfer these files to the JNIOR. You then run the Certificate Manager on the JNIOR to properly register the JNIOR.

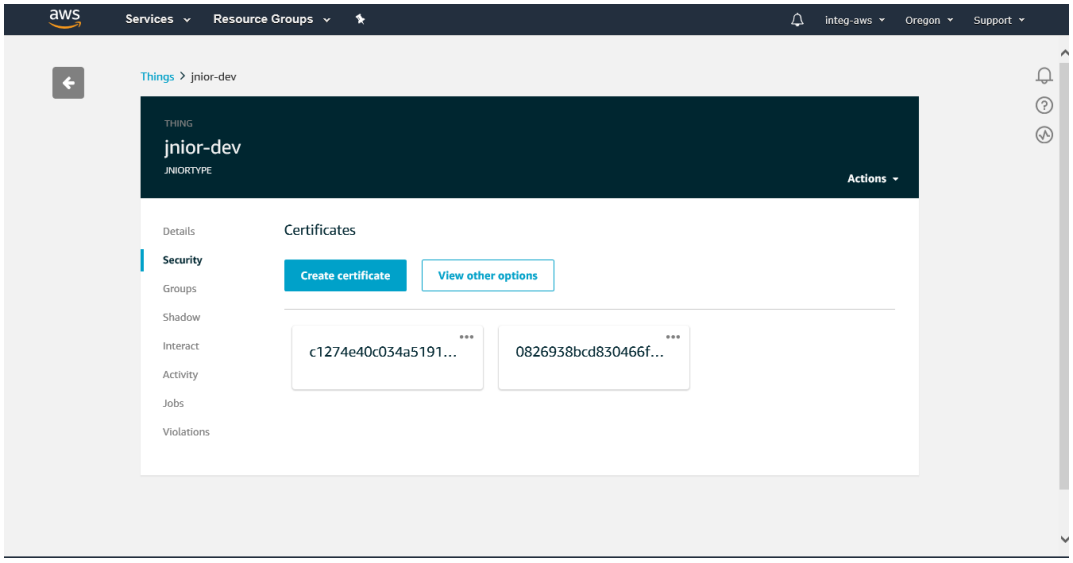
Below is a picture from the AWS Internet of Things Device Management web page for the INTEG account. You can see that we have two Things.

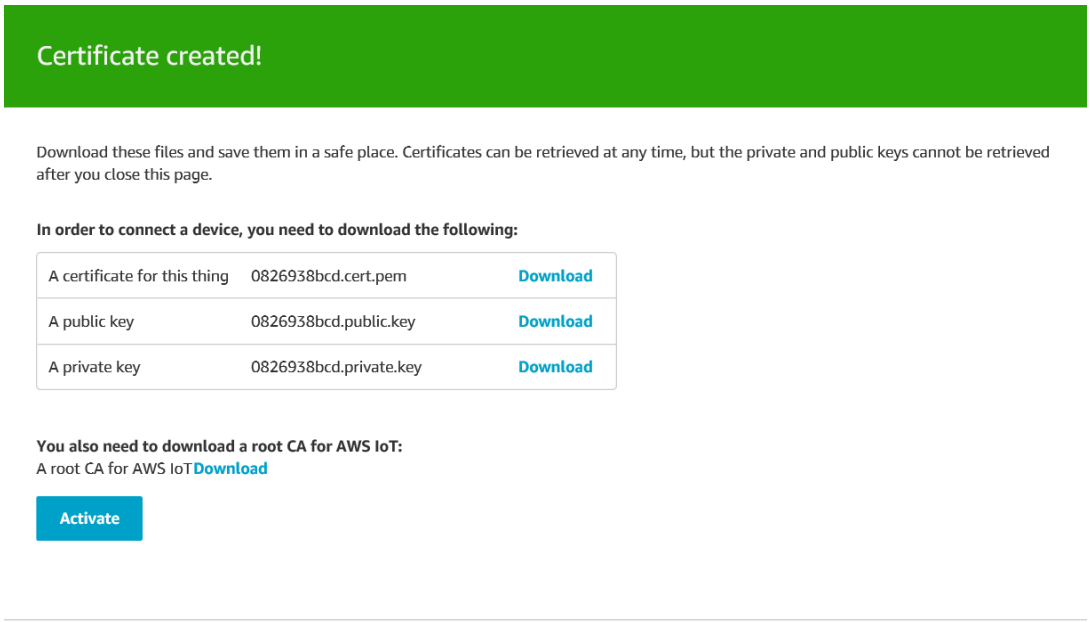


Below is a screen picture of the jnior-dev Thing.



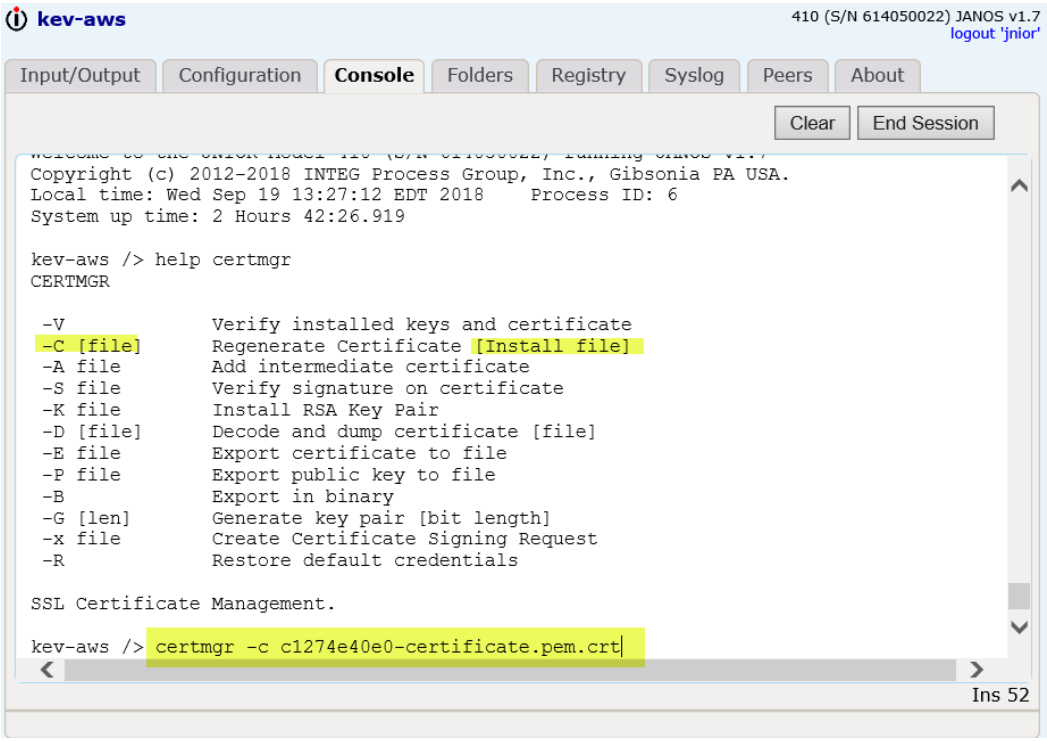
By clicking on the Security link and then Create certifcate, the three security files will be created as shown in the following two screen pictures.

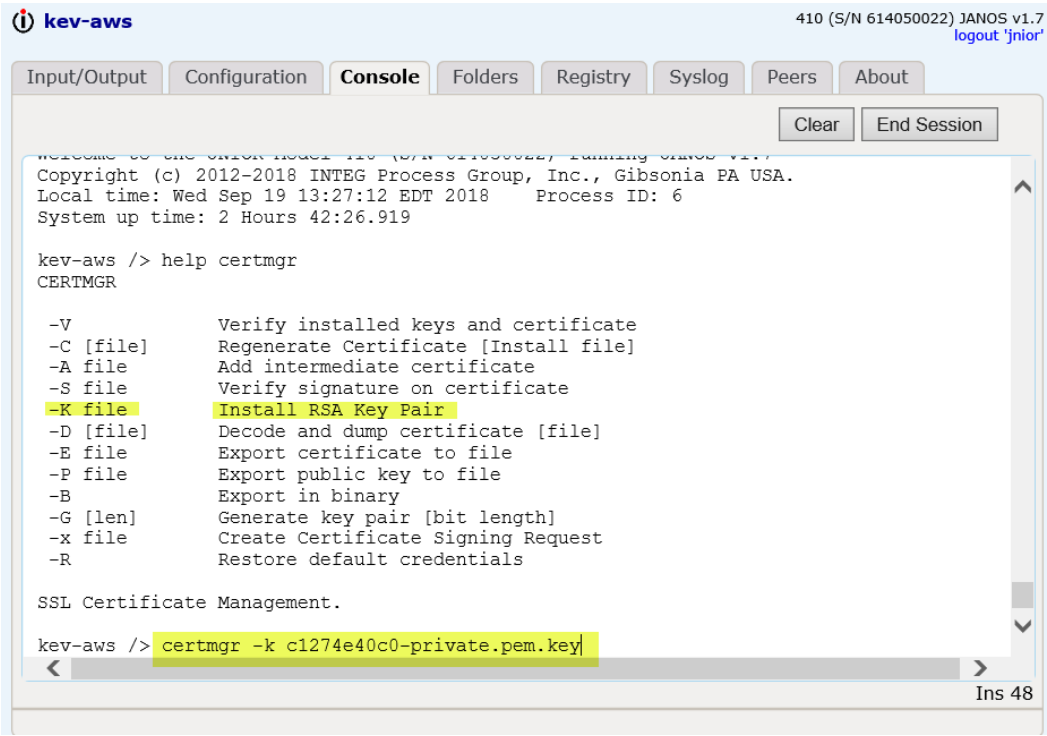




Using Windows FTP or the JNIOR web page, the first and third security files shown above (the certificate and private key) should be transferred to the JNIOR root directory.

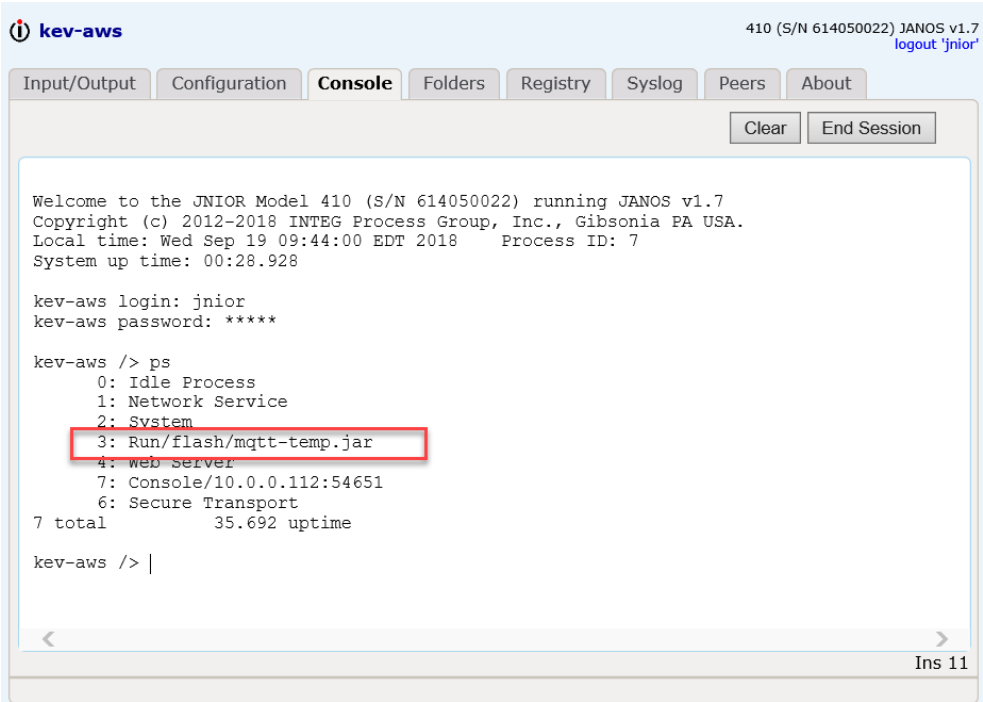
Once the files have been transferred, the security files can be ingested in the JNIOR using the certificate manager command as shown in the following two screen pictures.



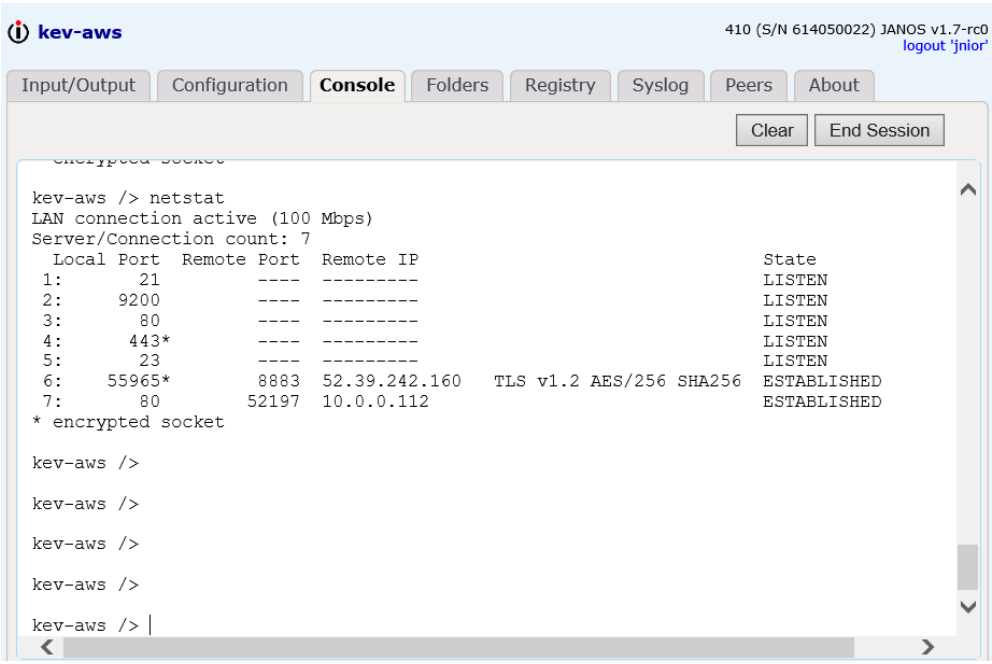


The next step is to load the JNIOR application for MQTT-AWS. Below is a picture of the JNIOR web page Console tab showing that the 'mqtt-temp' process is running.

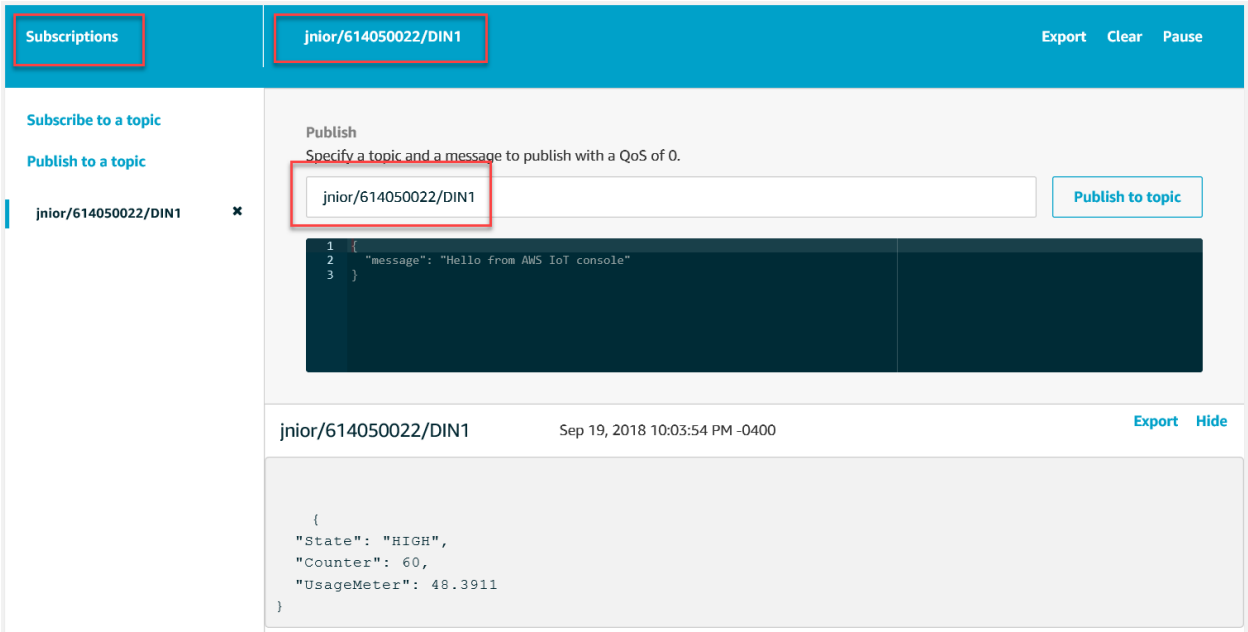
NOTE: Custom versions of the MQTT application can be developed to meet specific customer needs.

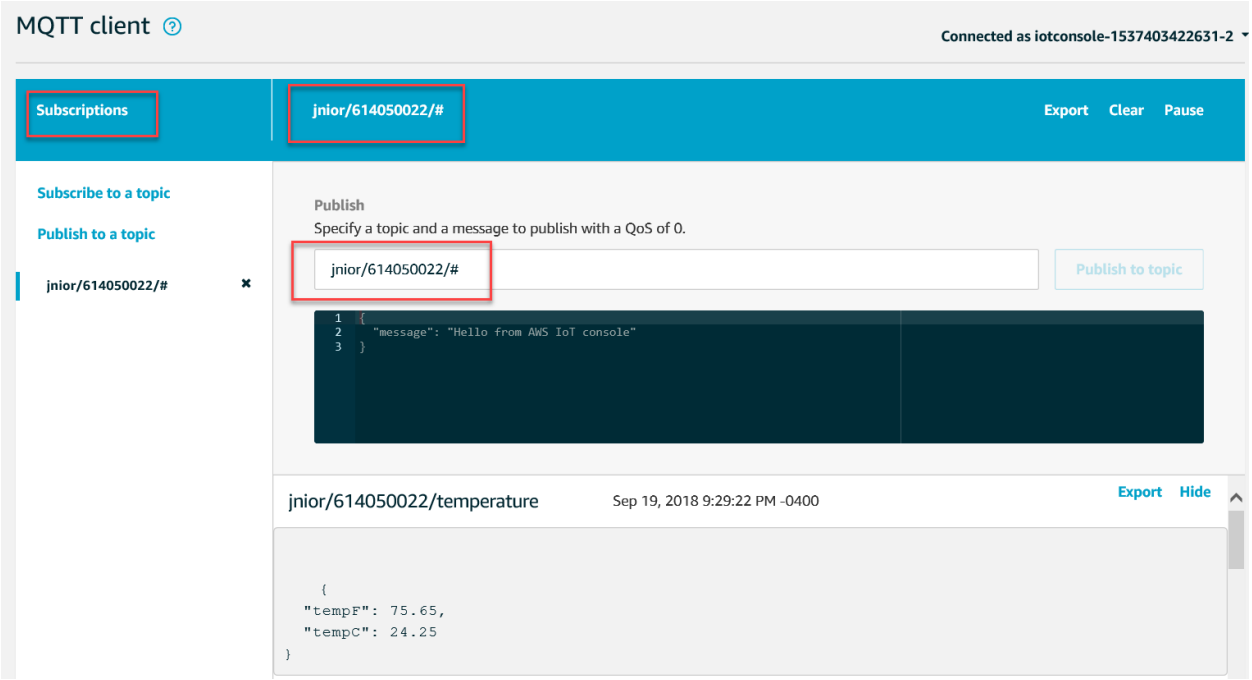


The screen picture below verifies that the JNIOR has made a secure connection to the AWS server.



Once the JNIOR side is completed and the AWS Thing created for the JNIOR, you can use the AWS test feature to 'subscribe' to the JNIOR topics. Below are several screen pictures first showing subscribing to a specific topic for JNIOR Input 1 and then the general wild card for all topics. Screen pictures are also provided of the payload for the various topics.





jnior/614050022/DIN4 Sep 19, 2018 8:50:20 PM -0400 Export Hide


```
{
  "State": "HIGH",
  "Counter": 5,
  "UsageMeter": 0.00879639
}
```


jnior/614050022/DIN4 Sep 19, 2018 8:59:37 PM -0400 Export Hide


```
{
  "State": "LOW",
  "Counter": 5,
  "UsageMeter": 0.163602
}
```


jnior/614050022/ROUT1 Sep 19, 2018 8:59:42 PM -0400 Export Hide

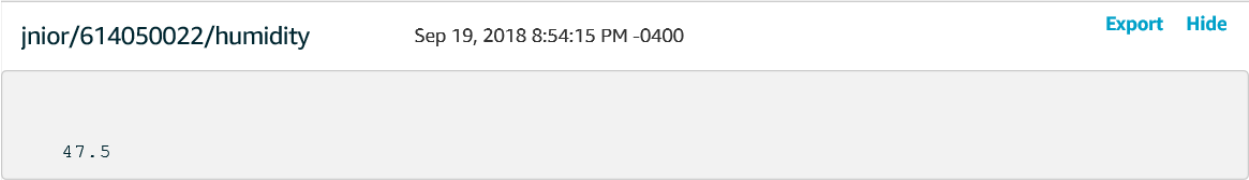

```
{
  "State": "HIGH",
  "UsageMeter": 5.53002
}
```


jnior/614050022/ROUT1 Sep 19, 2018 8:59:44 PM -0400 Export Hide


```
{
  "State": "LOW",
  "UsageMeter": 5.53054
}
```


jnior/614050022/temperature Sep 19, 2018 8:58:18 PM -0400 Export Hide


```
{
  "tempF": 75.65,
  "tempC": 24.25
}
```



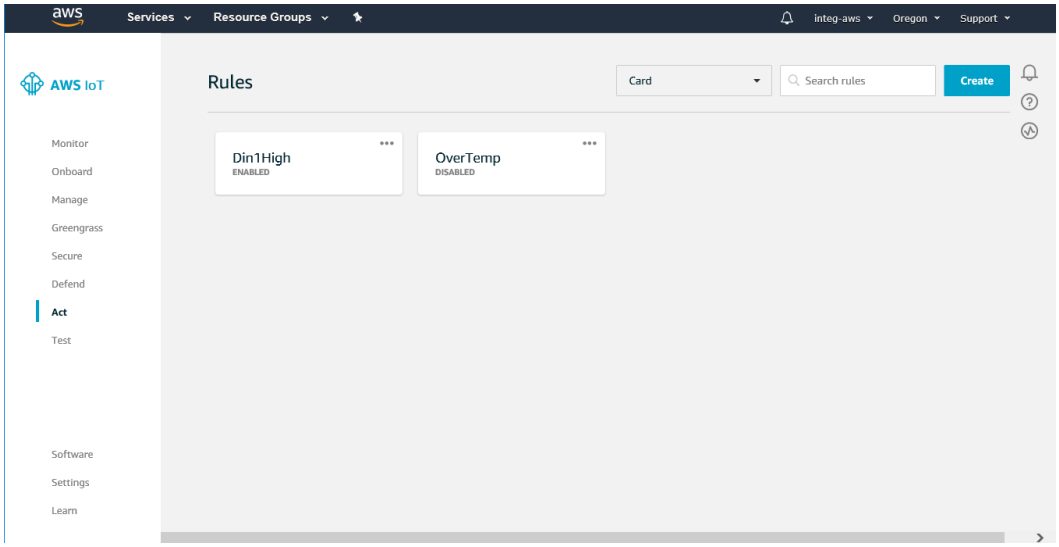
AWS Actions and Notifications

You can then utilize the AWS IoT functionality with their Simple Nework Notification service to monitor and react to various JNIOR I/O signals to implement a variety of actions including sending a text and email when certain parameters are met.

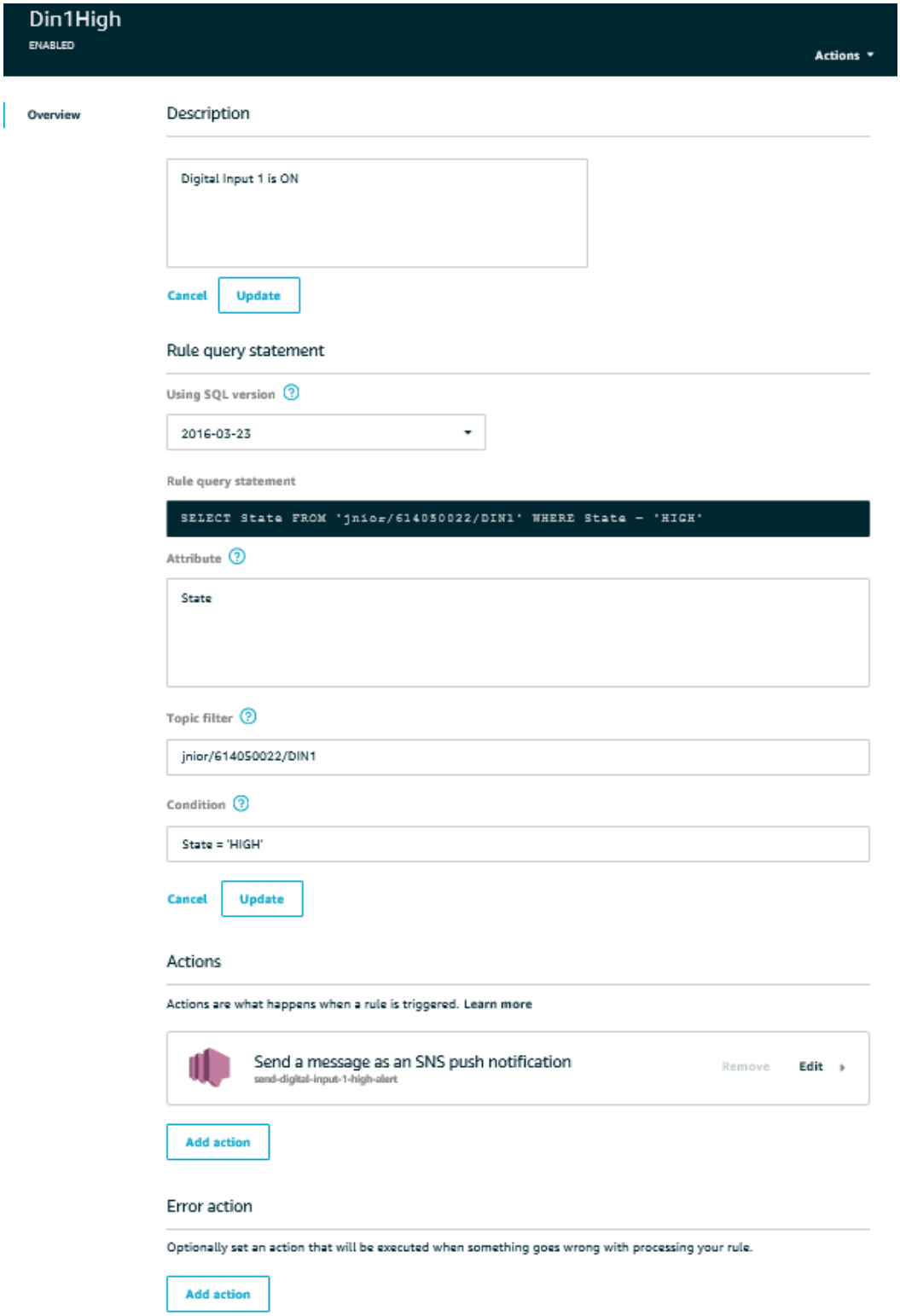
On the AWS IoT Act web page, you create various rules. The screen picture below indicates we have defined two rules so far.

Din1High is a rule related to digital input 1 going ‘high’.

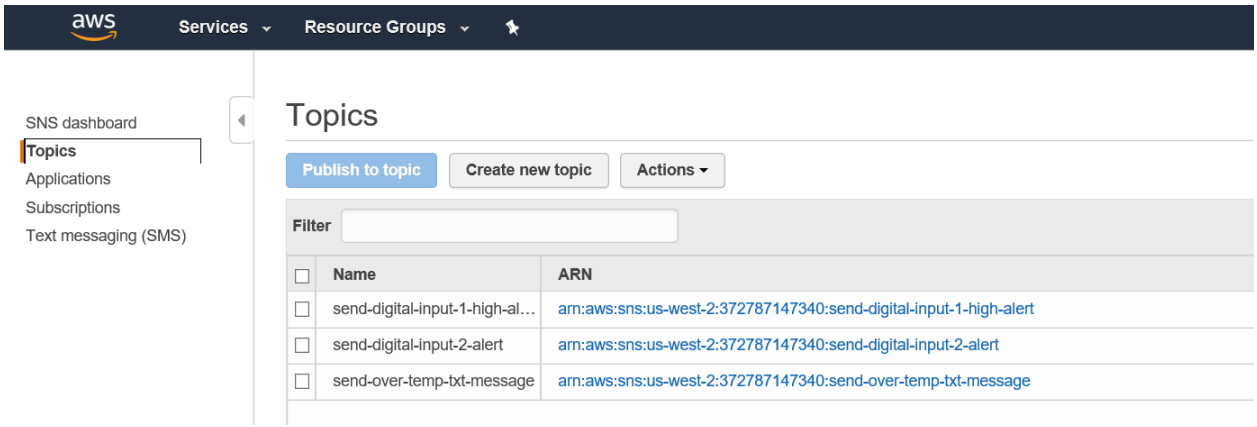
OverTemp is a rule related to the temperature reading going above a preset value.

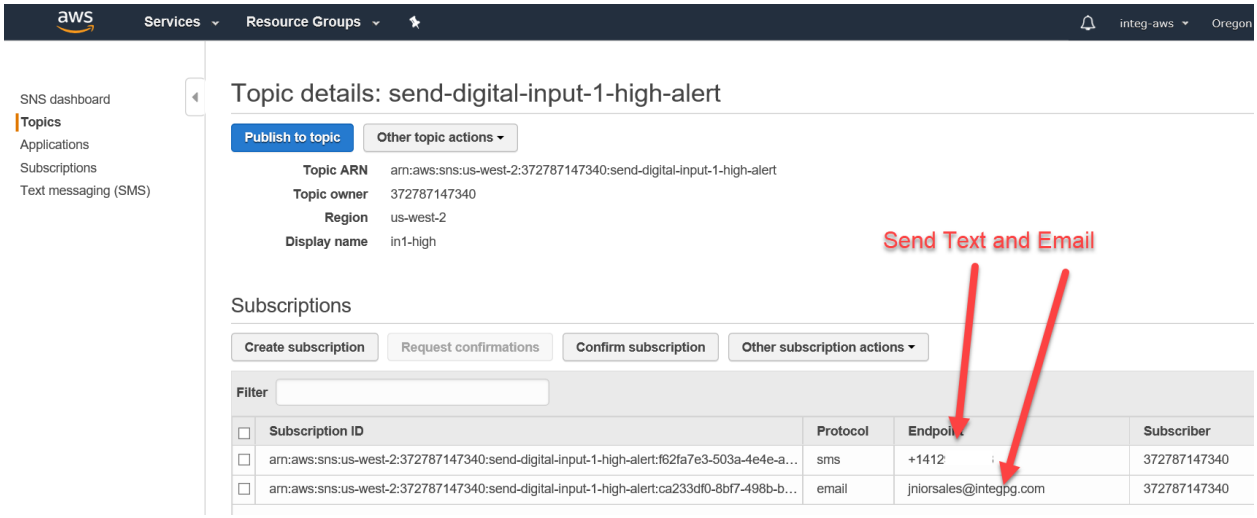


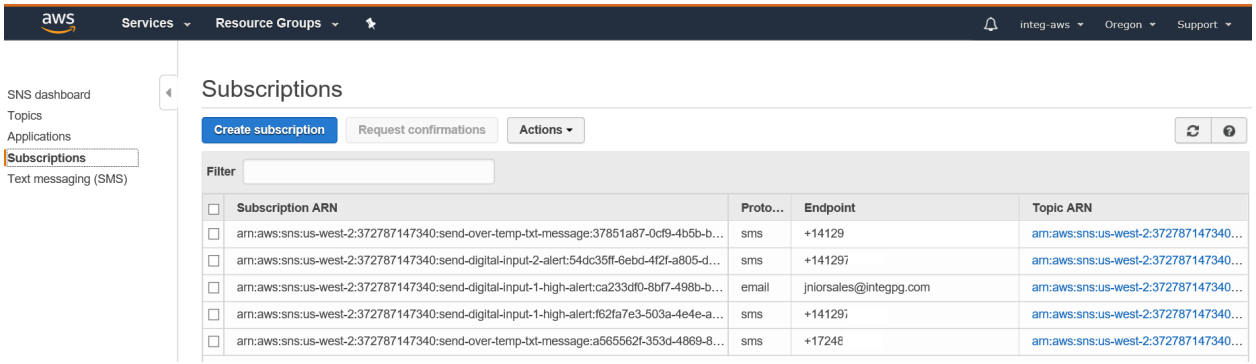
Below is a screen picture of the rule for Din1High. The ‘action’ is to send a text message using a SnS push notification.



Using the AWS Simple Network Notification, we can configure a topic and subscription that are used for the action above.







We can also create a rule where the action is triggered when the value of the temperature reading exceeds 75.5 degrees Farenheit.

RULE

OverTemp

ENABLED

Actions

Overview

Description Edit

Over 75.5

Rule query statement Edit

The source of the messages you want to process with this rule.

```
SELECT tempF FROM 'jnior/614050022/temperature' WHERE tempF > 75.5
```

Using SQL version 2016-03-23

Actions

Actions are what happens when a rule is triggered. **Learn more**

Send a message as an SNS push notification
send-over-temp-txt-message

Remove Edit

Add action

Error action

Optionally set an action that will be executed when something goes wrong with processing your rule.

Add action