# **Kerlink** communication is everything

Getting Started with Kerlink WAL-e and Amazon Web Services Getting Started Guide



## Getting Started with Kerlink WAL-e and Amazon Web Services

	Redaction	Validation	Approbation
Trigram	JCA	GBO	YDE
Date	2020-11-13		
Signature			

Version	Edits
1.0	Initial version
1.1	Update based on AWS feedback
1.2	Lambda code improvement
1.3	Typo corrections, clarifications add references

Reference	Description
[1]	https://wikikerlink.fr/wirnet-productline/doku.php?id=wiki:lora:aws



## Table of content

1 Prerequisites	
1.1 What you need	
1.2 Application overview	
2 Application Design	5
2.1 AWS IoT Core	5
2.1.1 Create rule	5
2.1.2 Add an action	7
2.1.3 Configure action	9
2.1.4 Finalize rule	
2.2 AWS Lambda	
2.2.1 Function code	
2.2.2 Deploy	
2.3 AWS Simple Notification Service	
2.3.1 Create topic	
2.3.2 Create subscription	
2.4 AWS Lambda permissions setup	
2.4.1 Lambda execution role	
2.4.2 Attach a new policy	
2.4.3 Attach policy to lambda role	
2.5 Follow application execution	
2.5.1 AWS IoT Core test	
2.5.2 AWS Cloudwatch	
2.5.3 AWS SNS publish	
3 Going further	

## **Table of figures**

Figure 1: WAL-e application on AWS overview	5
-igure 2: AWS IoT Core – Rules	6
-igure 3: AWS IoT Core - Create a rule	7
-igure 4: AWS IoT Core – Select an action	8
-igure 5: AWS IoT Core - Configure Lambda action	9
-igure 6: AWS lambda Create function	10
-igure 7: AWS IoT Core rule - Create rule	11



12
13
14
15
16
17
18
19
20
20
21
22
23
24
25
26
26



This guide will walk you through the creation of a cloud application for Kerlink WAL-e LoRaWAN sensor on Amazon Web Services (AWS).

### **1** Prerequisites

This tutorial is built on top of preceding Getting Started guides where you should have enabled your Kerlink WAL-e and routed its messages onto AWS IoT Core. Please refers to the following documents at [1]:

- Getting Started with Kerlink gateways AWS IoT Core for LoRaWAN
- Getting Started with Kerlink WAL-e AWS IoT Core for LoRaWAN

#### 1.1 What you need

To follow this guide, you need a Kerlink WAL-e configured and connected to AWS IoT Core. You also need an AWS account with sufficient permissions to use the following services

- AWS IoT Core
- AWS Lambda
- AWS Simple Notification Service

NOTE – all devices in your fleet must have credentials with privileges that authorize intended actions only, which include (but not limited to) AWS IoT MQTT actions such as publishing messages or subscribing to topics with specific scope and context. The specific permission policies can vary for your use cases. Identify the permission policies that best meet your business and security requirements.

For sample policies, refer to

<u>https://docs.aws.amazon.com/iot/latest/developerguide/example-iot-policies.html</u> . Also refer to <u>https://docs.aws.amazon.com/iot/latest/developerguide/security-best-</u> <u>practices.html</u>

#### **1.2 Application overview**

This guide will walk you through the process of creating an application which triggers a notification, either by email or by SMS, when the Kerlink WAL-e LoRaWAN sensor is activated. WAL-e has two modes of operation - It triggers an alarm either when the button is pressed or when a movement is detected.

No initial knowledge is required on LoRaWAN, nor on AWS services. It is recommended though to learn about AWS Policies and Permissions, as well as AWS pricing to avoid misuse of the services. All actions in this tutorial are covered by AWS free tier.



## **2** Application Design



Figure 1: WAL-e application on AWS overview

The application we are about to design relies on three AWS services, listed in the figure above. Messages are coming from the LoRaWAN Network into AWS IoT Core for LoRaWAN. We will define a rule to trigger an AWS Lambda function. That function will decode the payload sent by the sensor and trigger a notification on AWS SNS.

Finally, SNS will alert the user, forwarding the notification by email or by SMS.

#### 2.1 AWS IoT Core

#### 2.1.1 Create rule

First, create a rule in AWS IoT Core. Go to AWS IoT Core console at console.aws.amazon.com/iot, select "Act" >> "Rules", and click the "Create" button.



aws Services <b>T</b>	\$ (		Oregon V Support V
AWS IoT ×	AWS IoT > Rules		à
Monitor	Rules		Create
Onboard	Search rules	Q	
▶ Manage	Name	Status	
Greengrass     Secure		Enabled	•••
▶ Defend		Enabled	
▼ Act Rules Destinations			
Test			
Feedback English (US) ▼ © 200	) 8 - 2020. Amazon Web Services. Inc. or its affiliate	s. All rights reserved.	Privacy Policy Terms of Use
	Figure 2: AWS IoT Core – R	ules	

Give the rule a name and a description. Then, write the Rule query statement as follows:

SELECT \* FROM 'uplinks/wale'

Where uplinks/wale is the AWS IoT Core topic the messages are published to. This will select all the messages from that topic to run this rule.



Create a rule to evaluate messages sent by your things and specify what to do when a message is received (for example, write data to a DynamoDB table or invoke a Lambda function). Name wale_demo Description Demonstration rule for Kertlink WAL-e Rule query statement Indicate the source of the messages you want to process with this rule. Using SQL version 2016-03-23 Rule query statement	
wale_demo   Description   Demonstration rule for Kerlink WAL-e     Rule query statement   Indicate the source of the messages you want to process with this rule.   Using SQL version   2016-03-23     Rule query statement	
Description Demonstration rule for Kerlink WAL-e Rule query statement Indicate the source of the messages you want to process with this rule. Using SQL version 2016-03-23 Rule query statement	
Demonstration rule for Kerlink WAL-e Rule query statement Indicate the source of the messages you want to process with this rule. Using SQL version 2016-03-23 Rule query statement	
Rule query statement Indicate the source of the messages you want to process with this rule. Using SQL version 2016-03-23 • Rule query statement	
Rule query statement	
SELECT <attribute> FROM <topic filter=""> WHERE <condition>. For example: SELECT temperature FROM 'iot/topic' WHERE temperature &gt; flearn more, see AWS IoT SQL Reference.</condition></topic></attribute>	50. To
Set one or more actions Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occ when messages arrive, like storing them in a database, invoking cloud functions, or sending notifications. (*.required) Add action	cur
Error action Optionally set an action that will be executed when something goes wrong with processing your rule.	

#### Figure 3: AWS IoT Core - Create a rule

Next step is to add an action in that rule. Click "Add action" button.

#### 2.1.2 Add an action

Select the "Send a message to a Lambda function" action in the list.



AWS IoT > Rules >

Select an action				
Select an actio	ı			
•	Insert a message into a DynamoDB table DYNAMODB			
•	Split message into multiple columns of a DynamoDB table (DynamoDBv2)			
•	Send a message to a Lambda function			
0	Send a message as an SNS push notification			
0 🧧	Send a message to an SQS queue sqs			
0	Send a message to an Amazon Kinesis Stream			
0	Republish a message to an AWS IoT topic AWS IOT REPUBLISH			
0	Store a message in an Amazon S3 bucket s3			
0	Send a message to an Amazon Kinesis Firehose stream AMAZON KINESIS FIREHOSE			
· L	Send message data to CloudWatch metric			
• L	Change the state of a CloudWatch alarm CLOUDWATCH ALARMS			
• 🎚	Send message data to CloudWatch logs			
•	Send a message to the Amazon Elasticsearch Service			
O every	Send a message to a Salesforce IoT Input Stream SALESFORCE IOT			
0	Send a message to IoT Analytics			
<ul> <li>%</li> </ul>	Send a message to an IoT Events Input			
୍ କ୍ଷି	Send message data to asset properties in AWS IoT SiteWise			
0 🐐	Start a Step Functions state machine execution STEP FUNCTIONS			
ः	Send a message to a downstream HTTPS endpoint			
	Write a message into a Timestream table TIMESTREAM			
Cancel	Configure action			

Figure 4: AWS IoT Core – Select an action



#### 2.1.3 Configure action

Continue with clicking the "Configure Action" button at the bottom of the page.

AWS IoT > Rules >	
Configure action	
Send a message to a Lambda function	
We'll set <b>the permissions</b> on the Lambda function for you.	Create a new Lambda function
No lambda function selected	Select
Cancel	Add action

Figure 5: AWS IoT Core - Configure Lambda action

You need to create a lambda function for that action. Select "Create a new Lambda function" link. AWS interface will redirect you to the AWS Lambda page. Refer to Section 2.2 to configure your function.



aws	Services 🔻	\$ <b>(</b>	▼ Oregon ▼	Support 🔻
=	Lambda > Functions > Create funct	ion		^ (i)
	Create function Info			
	Choose one of the following options to	o create your function.		
	Author from scratch • Start with a simple Hello World example.	Use a blueprint O Build a Lambda application from sample code and configuration presets for common use cases.	Browse serverless app repository Deploy a sample Lambda application from the AWS Serverless Application Repository.	
	Basic information			
	Function name Enter a name that describes the purpose of	your function.		
	wale_decoder			
	Use only letters, numbers, hyphens, or under	rscores with no spaces.		
	Runtime Info Choose the language to use to write your fur	nction.		
	Python 3.8		•	
	Permissions late			
	By default, Lambda will create an execution default role later when adding triggers.	role with permissions to upload logs to Amazon C	loudWatch Logs. You can customize this	
	• Change default execution role			
	Advanced settings			
			Cancel Create function	,
Feedback	English (US) 🔻		Privacy Policy	Terms of Use
	© 2008	- 2020, Amazon Web Services, Inc. or its affiliates. All rights	reserved.	



For the moment, simply create a Python 3.8 function from scratch.

When the lambda is created, you can return to the AWS IoT Core Rule "Configure Action" (Figure 5) and click "Select" and eventually "Refresh" to select your freshly created function. Finish this procedure clicking "Add action" button.

#### 2.1.4 Finalize rule

Finally, once your action is added to the rule, create the rule on AWS IoT Core using the "Create Rule" button at the bottom of the page.



#### Set one or more actions

Select one or more actions to happen when the above rule is matched by an inbound message. Actions define additional activities that occur when messages arrive, like storing them in a database, invoking cloud functions, or sending notifications. (\*.required)

Þ	Send a message to a Lambda functio	n		Remove	Edit	•
Add ac	tion					
Error act Optionally Add ac	ion set an action that will be executed when somethin tion	ig goes wrong	with processing your rule.			
Tags Apply tags AWS resou	to your resources to help organize and identify the rces.	em. A tag cons	sts of a case-sensitive key-value pair. Lear	n more about	tagging	) your
Tag name	Provide a tag name, e.g. Manufacturer	Value	Provide a tag value, e.g. Acme-Corporat	ion	Cl	ear
Add an	other					
Cancel					Create r	ule

Figure 7: AWS IoT Core rule - Create rule

The rule is now created. You can now switch back to AWS Lambda interface to write the decoding function.



	aws	Services 🔻	\$ (	▼ Oregon ▼ Support ▼	
≡	Ø	Success Successfully created rule.		× ©	
		AWS IOT > Rules			
		Rules		Create	
		Search rules	Q		
		Name	Status		
			Enabled		
			Enabled	•••	
		vale_demo		***	

*Figure 8: AWS IOT Core - Successfully created rule* 

#### 2.2 AWS Lambda

Browse back to AWS Lambda page to configure your function. Either go to the browser tab you left while configuring AWS IoT Core action or go to AWS console and select Lambda service and select your function.

Your AWS Lambda function should look like the figure below. Note the link with AWS IoT in the trigger part of the Designer graph.



aws	Services <b>V</b>	\$ <b>—</b>	▼ Oregon ▼ Support ▼
=	Lambda > Functions > wale_deco	der ARN - 🗇 arn:aws:lambda:us-west-2:	::function:wale_decoder
	wale_decoder Throt	tle Qualifiers ▼ Actions ▼ Select	t a test event 🔻 Test
	Configuration Permissions	Monitoring	
	▼ Designer		
	AWS IoT + Add trigger	wale_decoder	+ Add destination
	Function code Info	Tools Window Test <b>*</b> Deploy	Deploy Actions V
	wale_decoder -/ <ul> <li>wale_function.py</li> <li>ambda_function.py</li> </ul> <ul> <li>ambda_function.py</li> </ul>	<pre>Iambda_functior × (+) I import json 2 3 def lambda_handler(event, context):</pre>	
Feedback	English (US) 🔻	© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.	Privacy Policy Terms of Use

#### Figure 9: AWS Lambda - empty function

#### 2.2.1 Function code

Browse to the Function code part of the page. There is a code editor to write the instructions of your function, with a template of Python code.

Copy and paste the following code snippet to the lambda\_function.py code area.

import json
import base64
import struct
import boto3
<pre>def lambda_handler(event, context):</pre>
# FPort = 2 is required
if event['ApplicationId'] != 2:
return



Figure 10: AWS Lambda – WAL-e decoder Python code

This code is hooked to an event, triggered by the previously defined AWS IoT Core rule. The event argument is a Python dictionary from the JSON received from AWS IoT Core. The code first filters on LoRaWAN FPort 2, then decodes the base64 payload data and unpacks each value into a Python dictionary named body.

The event dictionary keys are based on AWS IoT Core for LoRaWAN service. Depending on the LoraWAN Network Server used, those keys may be different and should be adapted.

We then use AWS boto3 Python module to publish the decoded elements to an AWS SNS topic named "IoT-to-email", adding an email subject and an email body containing the DevEUI of the device which triggered the lambda.

Note that you need to change the AWS SNS topic ARN with your own. See Section 2.3.1 to create that topic.

Note that you will need to add permission to your lambda function to publish on that topic. Refer to Section 2.4 to add the permission required.

To use SMS instead of email notification, as well as other functionalities of the boto3 module, please refer to the <u>boto3 documentation</u>.

#### 2.2.2 Deploy



Function code Info		Actions   Deploy
▲ File Edit Find View Go	Tools Window Test <b>*</b> Deploy	8 <b>\$</b>
te wale_decoder -// &* Imbda_function.py	<pre>Import json import json import json import struct import base64 import boto3  def lambda_handler(event, context):     # FPort = 2 is required     if event['ApplicationId'] != 2:         return  data = base64.b64decode(event['PayloadData']) values = struct.unpack_from("<hhhhh", 'humidity',="" 'temp',="" 'voltage')="" ,<br="" ['event',="" body="dict(zip(keys," body.update(event)="" client="boto3.client('sns')" data)="" keys="('version'," response="client.publish(" topicarn='arn:awsisns:us-west-2: ::IoT-to-email     Message = "Motion detected"     subject = "Motion detected" }&lt;/pre&gt;&lt;/th&gt;&lt;th&gt;' values))="">LoRaWAN']['DevEUI'][8:]), 13:24 Python Spaces:4 ✿</hhhhh",></pre>	

Figure 11: AWS Lambda – Deploy

Hit the "Deploy" button to enable the function.

The code is ready. The next step is to configure AWS Simple Notification Service to receive an email when an uplink occurs.

#### 2.3 AWS Simple Notification Service

Go back to the AWS console home and look for Simple Notification Service. The service starts with a landing page.



aws	Services ▼		\$	▼ Oregon ▼ Support ▼
=	Application Integration			
	Amazon Simp Notification S	le ervice	Create topic	
	Pub/sub messa microservices a applications.	aging for and serverless	Topic name A topic is a message channel. When you you to a topic, it fans out the message to all s endpoints. MyTopic	publish a message subscribed
	Amazon SNS is a highly available, durable, sec that enables you to decouple microservices, di applications. Amazon SNS provides topics for messaging.	ure, fully managed pub/sub messaging service stributed systems, and event-driven serverless high-throughput, push-based, many-to-many	Next step	
			Pricing	
	Benefits and features		Amazon SNS has no upfront costs.	You pay based
	Reliably deliver messages with durability	Automatically scale your workload	on the number of messages you pu number of messages you deliver, an API calls for managing topics and s Delivery pricing varies by endpoint	iblish, the nd any additional ubscriptions. type.
	zone message storage to provide high message durability. Amazon SNS reliably delivers messages to valid	AWS cloud to dynamically scale with your application. Amazon SNS is a fully managed service, taking care of the hear. If the plant dynamical	Learn more	
	AWS endpoints, such as Amazon SQS queues and AWS Lambda functions.	the neavy lifting related to capacity planning, provisioning, monitoring, and patching.	Documentation 🖸	
<	Simplify your architecture	Keen messages private and	Developer Guide	~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~
Feedback	English (US) 🔻	© 2008 - 2020, Amazo	n Web Services, Inc. or its affiliates. All rights reser	ved. Privacy Policy Terms of Use

Figure 12: AWS SNS - landing page

Type in the topic we chose in the AWS Lambda code and click "Next step".

#### 2.3.1 Create topic



aws Services 🔻	\$ <b>(11)</b>	🔹 Oregon 🔻 Support 🔻
Amazon SNS ×	Amazon SNS > Topics > Create topic	٥
Dashboard Topics	Create topic	
Subscriptions	Details	
<ul> <li>Mobile</li> <li>Push notifications</li> <li>Text messaging (SM5)</li> </ul>	Strictly-preserved message ordering         • Exactly-once message delivery         • High throughput, up to 300 publishes/second         • Subscription protocols: SQS	
	Name IoT-to-email Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).	
	Display name - optional           To use this topic with SMS subscriptions, enter a display name. Only the first 10 characters are displayed in an SMS message. Info           My Topic           Maximum 100 characters, including hyphens (-) and underscores ().	
	Encryption - optional Amazon SNS provides in-transit encryption by default. Enabling server-side encryption adds at-rest encryption to your topic.	
	Access policy - optional This policy defines who can access your topic. By default, only the topic owner can publish or subscribe to the topic. Info	
	Delivery retry policy (HTTP/S) - optional The policy defines how Amazon SNS retries failed deliveries to HTTP/S endpoints. To modify the default settings, expand this section. Info	
	Delivery status logging - optional     These settings configure the logging of message delivery status to CloudWatch Logs. Info	
	► Tags - optional A tag is a metadata tabe! that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to se track your costs. Learn more C	arch and filter your topics and
	c	ancel Create topic
Feedback English (US) 🔻	© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All right	s reserved. Privacy Policy Terms of Use

Figure 13: AWS SNS - Create topic

Simply check the name and hit "Create topic" button at the bottom of the page. Refer to AWS SNS documentation to know more about the other options.

Once the topic created, note the ARN for your lambda function (Refer to Section 2.2.1).



aws Services 🔻		
Amazon SNS ×	Topic IoT-to-email created successfully.     You can create subscriptions and send messages to them from this topic.	١
Dashboard Topics Subscriptions V Mobile	Amazon SNS > Topics > IoT-to-email         IOT-to-email         Edit       Delete	
Push notifications Text messaging (SMS)	Details	
	Name       Display name         IoT-to-email       -         ARN       Topic owner         arn:aws:sns:us-west-2:       :loT-to-email         Type       Standard         Subscriptions       Access policy       Delivery retry policy (HTTP/S)       Delivery status logging       Encryption         Tags	
	Subscriptions (0)       Edit     Delete       Request confirmation     Confirm subscription	
	Q Search O	
Feedback English (US) ▼	D     T     Endpoint     Statur     C     Statur     C     D	lse

Figure 14: AWS SNS - topic created successfully

#### 2.3.2 Create subscription

On the topic page, there is a subscription section. Hit the "Create subscription". This will setup the email notification for the topic we just created.



aws	Services 🔻		\$ <b>(</b>		Oregon 🔻	Support 🔻
=	Amazon SNS > Subscriptions > Create subscription					٩
	Create subscription					
	Details					
	Topic ARN					
	Q arn:aws:sns:us-west-2:	×				
	Protocol The type of endpoint to subscribe					
	Email	•				
	Endpoint An email address that can receive notifications from Amazon SNS.					
	test@example.com					
	(1) After your subscription is created, you must confirm it. Info					
	• C headada filmenting antiquit					
	Subscription filter policy - optional     This policy filters the messages that a subscriber receives. Info					
	Redrive policy (dead-letter queue) - optional Send undeliverable messages to a dead-letter queue. Info					
			Cancel	Create subscription	]	
Feedback	English (US) 🔻	© 2008 - 2020, Amazon Web S	Services, Inc. or i	ts affiliates. All rights reserved.	Privacy Policy	Terms of Use

Figure 15: AWS SNS - Create subscription

Choose "Email" under the Protocol list and enter the e-mail of the notification recipient in the Endpoint field.

You can also select "SMS" if you prefer this notification method and add the phone number to be notified. Service charges may apply.

Finally, hit "Create subscription" button.



aws Services <b>v</b>			\$ <b></b>	Oregon ▼ Support ▼
Amazon SNS	×	Subscription to IoT-to-email created successfully. The ARN of the subscription is arn:aws:sns:us-west-2:	:loT-to-email:7f531c6cf7af2808	4f55.
Dashboard Topics		Amazon SNS > Topics > IoT-to-email > Subscription:	7f531c6c- f7af28084f55	
Subscriptions		Subscription: 7f531c6c-	-f7af28084f55	
<ul> <li>Mobile</li> <li>Push notifications</li> </ul>		Edit Delete		
Text messaging (SMS)		Details		
		ARN arn:aws:sns:us-west-2: email:7f531c6c- Endpoint @kerlink.fr Topic IoT-to-email	Status Pending confirmation Protocol EMAIL	
		Subscription filter policy Redrive policy (dead-lett	er queue)	
		Subscription filter policy This policy filters the messages that a subscriber receives. Info		
		No filter policy cor	nfigured for this subscription.	
		To apply a filter p	policy, edit this subscription.	
Feedback English (US) 🔻		© 2008 -	2020, Amazon Web Services, Inc. or its affiliates. All rights reserved.	Privacy Policy Terms of Use

Figure 16: AWS SNS - Subscription created

Note that the subscribed email needs to be confirmed. You will receive a confirmation email immediately after creating the subscription. Click on the "Confirm subscription" link.

webservices	Simple Notification Service
Subscription confirmed!	
You have subscribed @kerlink.fr IoT-to-email.	to the topic:
Your subscription's id is:	
arn:aws:sns:us-west-2: -f7af28084f55	:IoT-to-email:7f531c6c
If it was not your intention to subs	cribe, <u>click here to unsubscribe</u> .

Figure 17: AWS SNS - Subscription confirmed

#### 2.4 AWS Lambda permissions setup

The final step is to allow the lambda function we created to post to the SNS topic.



#### 2.4.1 Lambda execution role

To add permission to the lambda function, go to AWS Lambda, and select the function. Switch to the "Permissions" tab of the function and click on the role name under "Execution role".

aws Services V		\$ <b>(</b>	Oregon 🔻	Support 🔻
■ Lambda > Functions > wale_de	coder ARN - 🗇 a	rn:aws:lambda:us-west-2:	:function:wale_decod	er î
wale_decoder	Throttle Qualifiers <b>v</b>	Actions  Select a test	event <b>Test</b>	
Configuration Permissions	Monitoring			
Execution role			Edit	
Role name wale_decoder-role-l2dfopOc				
Resource summary		[	View role document	
Amazon CloudWatch Loge 3 actions, 2 resources	ŝ			
To view the resources and actions	that your function has permission to access, choose	a service.		
By action By resource				
Resource		Actions		
arn:aws:logs:us-west-2:	*	Allow: logs:C	reateLogGroup	
arn:aws:logs:us-west-2:	:log-group:/aws/lambda/wale_decoder:*	Allow: logs:C Allow: logs:P	reateLogStream utLogEvents	
	Figure 18: AWS Lambda –	Permissions		

This link leads to AWS Identity and Access Management (IAM), on the role summary of the function.

#### 2.4.2 Attach a new policy



	Summary				D	elete role
Dashboard	Role ARN	arn:aws:iam::/	:role/service-r	ole/wale_decoder-role-l2	2dfop0c 省	
Access management	Role description	Edit				
Groups	Instance Profile ARNs	62				
Users	Path	/service-role/				
Roles	Creation time	2020-11-13 17:24	UTC+0100			
Policies	Last activity	2020-11-13 21:0	UTC+0100 (Today)			
Identity providers	Maximum session	1 hour Edit				
Account settings	duration					
Access reports	Permissions Trust relation	onships Tags	Access Advisor	Revoke sessions		
Access analyzer	<ul> <li>Permissions policies</li> </ul>	(1 policy applie	t)			
Archive rules		(i ponoj appno	-/			
Analyzers	Attach policies				Add inline	policy
Settings	Policy name 👻		Policy type 👻			
Credential report	AWSI ambdaBasicE	vecutionRole-7eb4	Managed policy			×
Organization activity						
Service control policies (SCPs)	<ul> <li>Permissions boundar</li> </ul>	y (not set)				
<b>Q</b> Search IAM						
/s account ID:						

Figure 19: AWS IAM - Role summary

Click "Attach policies" button under the Permissions tab.



aws	Servic	es 🔻	,		4	)	_	🔻 Global 🔻	Support 🔻
	Add permissions to wale, decoder-role-12dfop0c								
	Cro					~			
	Grea	ate p	oond	SY		U			
	Filter policies V Q Search			S v Q Search		g 620 results			
			Po	licy name 👻	Туре	Used as			
		•	Û	AdministratorAccess	Job function	Permissio	ns policy (1)	^	
		•	Û	AlexaForBusinessDeviceSetup	AWS managed	None			
		•	Û	AlexaForBusinessFullAccess	AWS managed	None			
		•	Ũ	AlexaForBusinessGatewayExecution	AWS managed	None			
		•	Û	AlexaForBusinessLifesizeDelegatedAccessPolicy	AWS managed	None			
		•	Û	AlexaForBusinessPolyDelegatedAccessPolicy	AWS managed	None			
		•	Û	AlexaForBusinessReadOnlyAccess	AWS managed	None			
		•	Û	AmazonAPIGatewayAdministrator	AWS managed	None			
		•	Û	AmazonAPIGatewayInvokeFullAccess	AWS managed	None			
		•	Û	AmazonAPIGatewayPushToCloudWatchLogs	AWS managed	None			
		•	Û	AmazonAppFlowFullAccess	AWS managed	None			
		•	Û	AmazonAppFlowReadOnlyAccess	AWS managed	None			
		•	Û	AmazonAppStreamFullAccess	AWS managed	None			
		•	Û	AmazonAppStreamReadOnlyAccess	AWS managed	None			
							Cancel	Attach policy	
Feedback	English	(US)	•		© 2008 - 2020, Amazon Web Services, In	ic. or its affiliates. ,	All rights reserved.	Privacy Policy	Terms of Use

Figure 20: AWS IAM - Add permissions

On the next page, click "Create policy" button to add a new policy for that role.



av	NS	Services 🔻			¢			🖌 Global 🔻 🚽	Support 🔻
		Create policy						1 2	
		A policy defines the AWS permissions that y Visual editor JSON	ou can assign to a user, g	group, or role. You can create and edit	a policy in the visual ed	itor and using JS	ON. Learn n Impor	nore t managed policy	
		Expand all   Collapse all							
		✓ SNS (1 action)					Clor	e Remove	
Documentation		Service     Actions     Resources     close	SNS Write Publish Specific All resources topic ③	am:aws:sns:us-west-2 Add ARN to restrict access	loT-to-email	EDIT O	Any	in this account	
		Request conditions	Specify request condition	ons (optional)					
						O Add	additional	permissions	
	Cha	aracter count: 163 of 6 144.					Cancel	Review polic	ey 🛛
Feedt	back	English (US) 🔻		© 2008 - 2021	0, Amazon Web Services, Inc.	or its affiliates. All righ	its reserved.	Privacy Policy	Terms of Use

Figure 21: AWS IAM - Create policy

Select the "SNS" service. On the actions list, tick "Publish" under "Write" actions. Finally, select a specific resource, giving the topic ARN you just created in AWS SNS. Hit the "Review policy" button on the bottom of the page.



aws	Services <b>▼</b>				\$ <b>—</b>		Global 🔻 Support 🔻	
	Create policy					1	2	
	Review policy Name* Description	SNS-publish-IoT-Io-email Use alphanumeric and '+=,@' chai Policy to allow a lambda functi Maximum 1000 characters. Use alph	acters. Maximum 128 characte on to publish to Stys topic <u>1</u> anumeric and '+=. @- <u>-</u> ' charad	rs. QT_to-email ters.			÷	
	Summary		Access level		Resource	Request conditio		
		Allow (1 of 242 services) SI	now remaining 241			Request contain		
		SNS	Limited: Write		TopicName   string like   IoT-to-email	None	<u>`````````````````````````````````````</u>	
		,					,	
* R	Required				Cancel	Previous	reate policy	
Feedback	English (US) 🔻			© 2008 - 2	2020, Amazon Web Services, Inc. or its affiliates. A	Il rights reserved. Priv	acy Policy Terms of Use	Ē,

Figure 22: AWS IAM - review policy

Give the policy a name and a description. Finally, click the "Create policy" button.

#### 2.4.3 Attach policy to lambda role



aws Services V	
Identity and Access Management (IAM)	Policies > SNS-publish-IoT-to-email Summary Delete policy
Dashboard	Policy ARN arn:aws:lam:: policy/SNS-publish-loT-to-email
	Description Policy to allow a lambda function to publish to SNS topic IoT-to-email
Groups	Permissions Policy usage Policy versions Access Advisor
Users	Permissions Policy usage Policy versions Access Advisor
Roles	✓ Permissions
Policies	Attach this policy to an IAM entity to apply its permissions to the entity. Learn more
Identity providers	Attach
Account settings	
	Filter: Filter V Q Search Showing 0 results
Access analyzer	
Archive rules	innine .
Analyzers	No results
Settings	Permissions boundaries
Credential report	
Organization activity	
Service control policies (SCPs)	
Q Search IAM	
AWS account ID:	
Feedback English (US) 🔻	© 2008 - 2020, Amazon Web Services, Inc. or its affiliates. All rights reserved. Privacy Policy Terms of Use

Figure 23: AWS IAM –Policy usage

Navigate to your newly created policy. Under the "Policy usage" tab, hit the "Attach" button.

aws	Services 🔻		\$ <b></b>	V Global V	▼ Support ▼			
	Attach policy Attach the policy to users, groups, or roles in your account							
	Filter: Filter ~ Q wale_decoder-role-l2dfop0c							
	✓ Name ▼			Туре 👻				
	wale_decoder-role-l2dfop	Dc		Role				
				Cancel	Attach policy			
Feedback	English (US) 🔻			Privacy Poli	cy Terms of Use			
		© 2008 - 2020, Amazon Web Ser	vices, Inc. or its affiliates. All rights reserved.					

Figure 24: AWS IAM - Attach policy



Search for the lambda function role in the search bar, tick its line and finally hit "Attach policy" button.

If you navigate back to the lambda function role, you will see that 2 policies are attached to the role.

#### **2.5 Follow application execution**

Your application is now all setup and ready to run. For each uplink message received on AWS IoT Core, an email will be sent by AWS SNS.

To troubleshoot or watch your application behaviour, several tools come handy.

#### 2.5.1 AWS IoT Core test

Navigate to the AWS IoT Core console. In the navigation pane on the left, click Test. This takes you to the MQTT Test client page, where you can subscribe to an IoT Core topic and watch the data flowing in. You can also manually publish data on any IoT Core topic.

#### 2.5.2 AWS Cloudwatch

Cloudwatch collects the logs of the lambda function. If the function raises an error, Cloudwatch logs will collect and display the function traces.

#### 2.5.3 AWS SNS publish

AWS SNS allows to manually publish a message onto a given topic. This will help investigating if notification emails go to your spam box.

#### **3 Going further**

This is it! You have designed and set-up a complete end-to-end application, from the LoRaWAN sensor up to the e-mail notification system.

You can imagine many other usages on top of this infrastructure, such as storing the data sent by the device into AWS S3 or DynamoDB, try the AWS IoT Core Analytics, and even reply back to the device to light an LED.

#### **End of Document**