Internet of Things

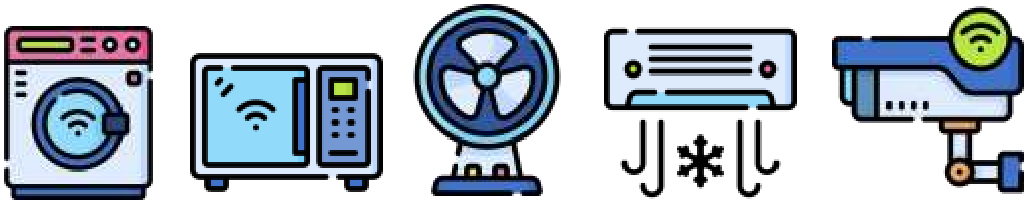
Connecting Things to AWS IoT Core

Abdallah El Ghamry

IoT Applications

Things

- “Things” are a generic set of entities, including smart devices, sensors, human beings, and any other object that is aware of its context and is able to communicate with other entities, making it accessible at anytime, anywhere.

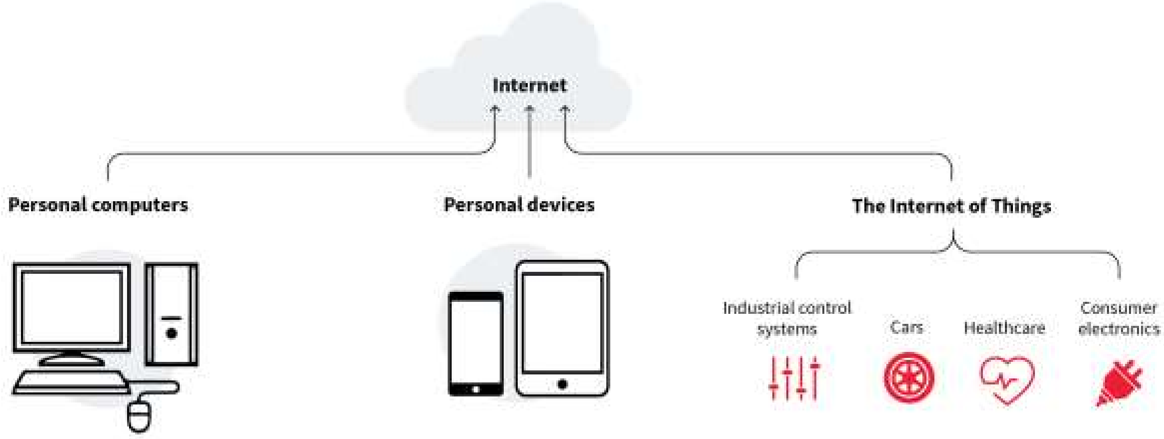


3

Things

4

Internet of Things

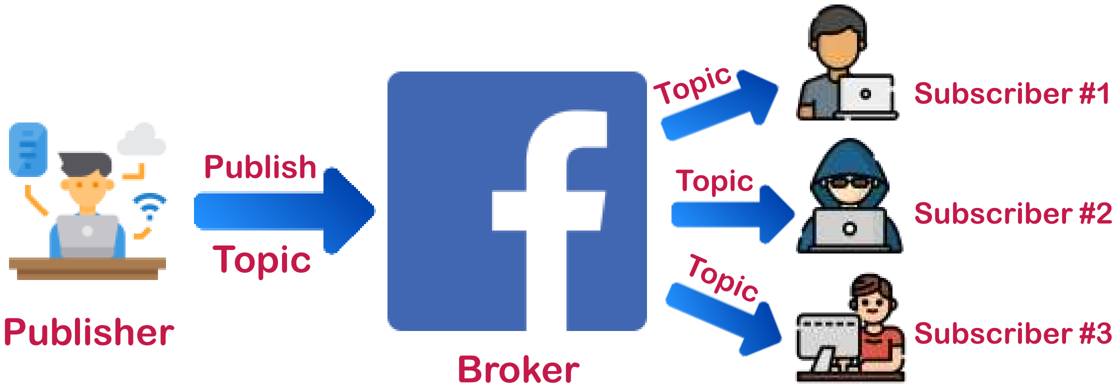


Things

- The Internet of Things (IoT) represents the network of physical objects "Things" that are integrated with sensors, software and other technologies for the purpose of exchanging data with other devices on the Internet.

MQTT: Informal Introduction

- An admin (publisher) can publish a new post (topic) on a Facebook page.
- Facebook (broker) will send that topic to subscribers who liked the page.

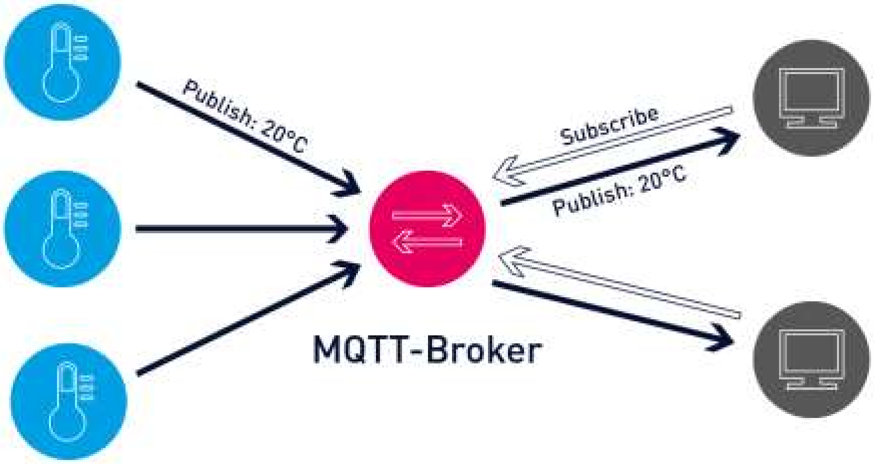


MQTT: Formal Introduction

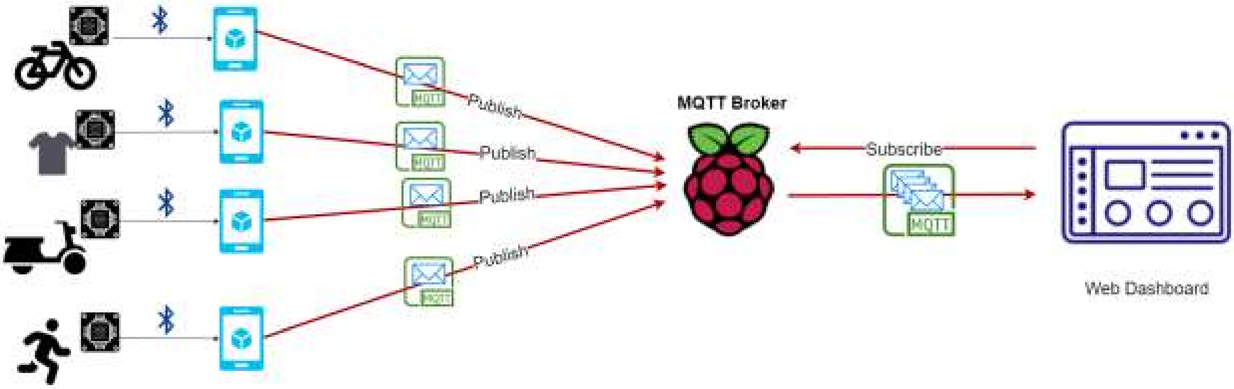
- MQTT stands for Message Queuing Telemetry Transport, a simple messaging protocol suitable for communication between IoT devices.
- MQTT is more and more becoming the standard messaging protocol for IoT messaging.
- MQTT was developed by IBM in 1999.
- MQTT is very lightweight.
- As all workload is done by the broker.

MQTT: Broker

- In MQTT, the clients (such as sensors, machines, and applications) do not directly communicate with each other but via a broker.



MQTT: Broker

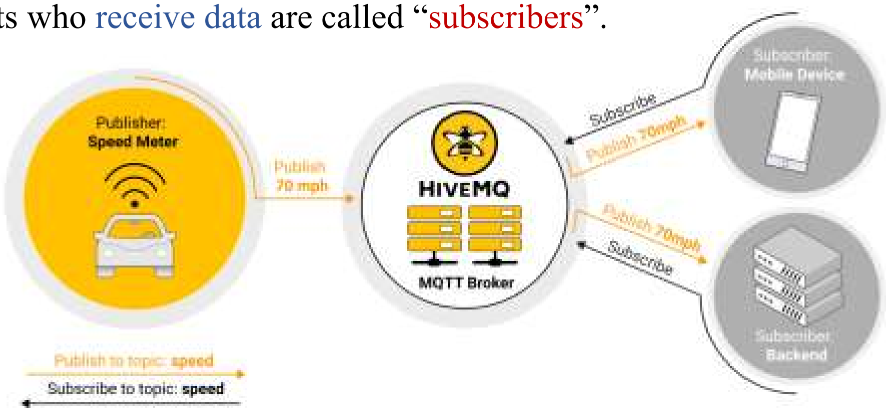


MQTT: Broker

- MQTT brokers are the heart of a connected IoT application.
- And just as functioning of the heart is critical for the human body, a reliable and performant MQTT broker is critical for IoT operations.
- The MQTT broker receives the data from the senders, filters the data packets, and forwards them to the receiving clients.

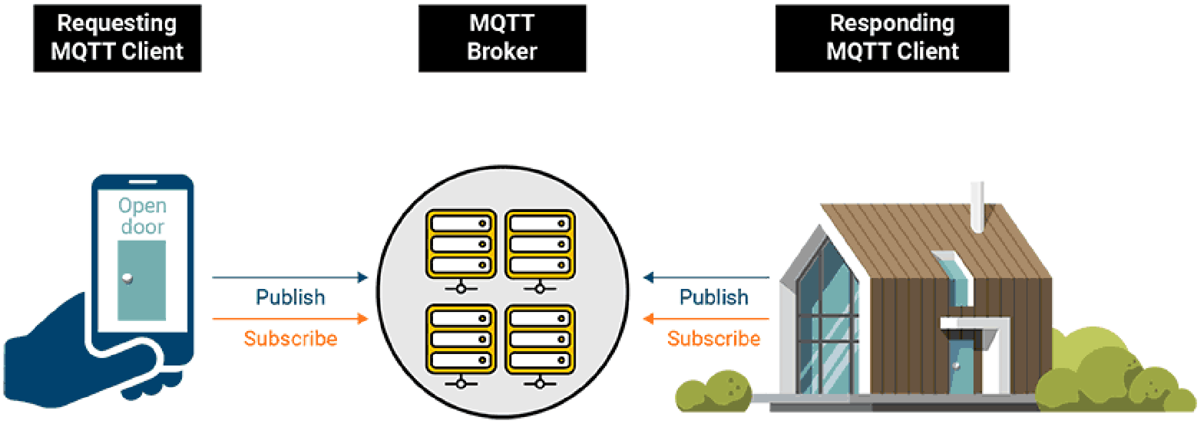
MQTT: Publishers & Subscribers

- In a publish and subscribe system, a device can publish a message on a topic, or it can be subscribed to a particular topic to receive messages
- Clients sending data are called "publishers".
- Clients who receive data are called "subscribers".



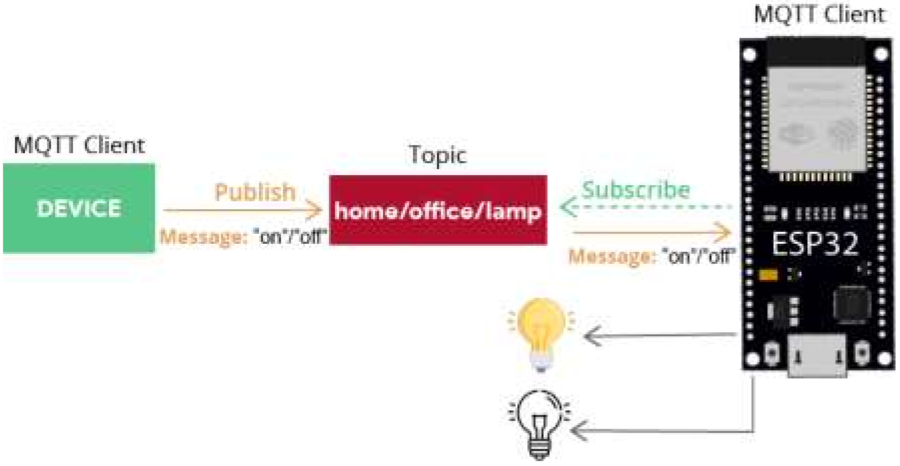
MQTT: Publishers & Subscribers

- An MQTT system enables receiving clients to become publishers as well.



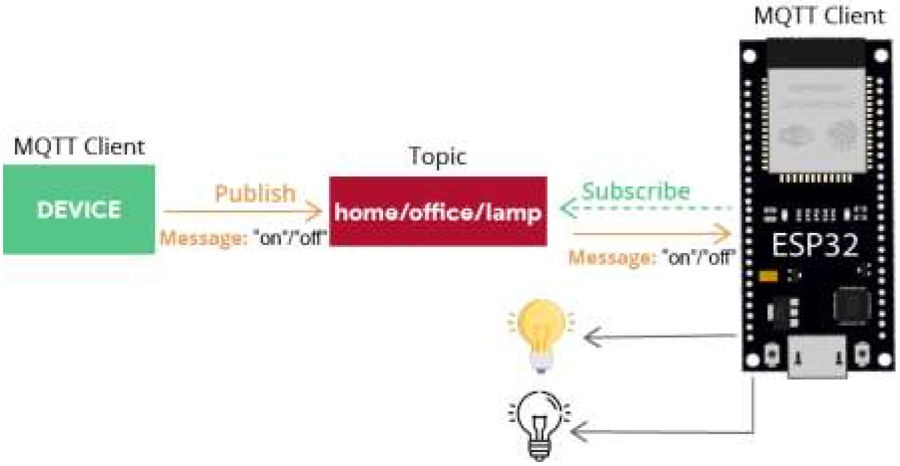
MQTT: Messages

- Messages are the data that you want to exchange between your devices.
- For example, a message can be a command or data like sensor readings.

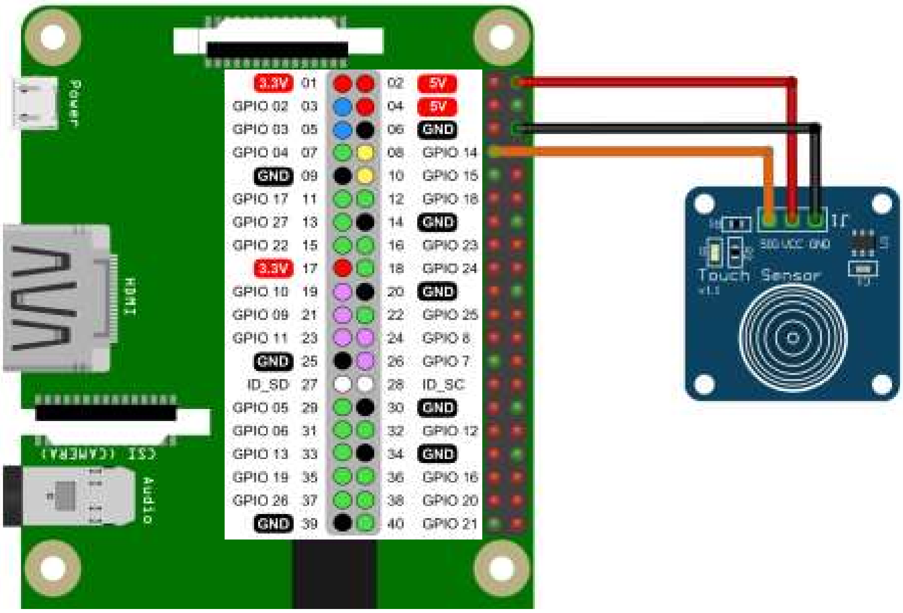


MQTT: Topics

- A topic is the way you register interest for incoming messages or how you specify where you want to publish the message.

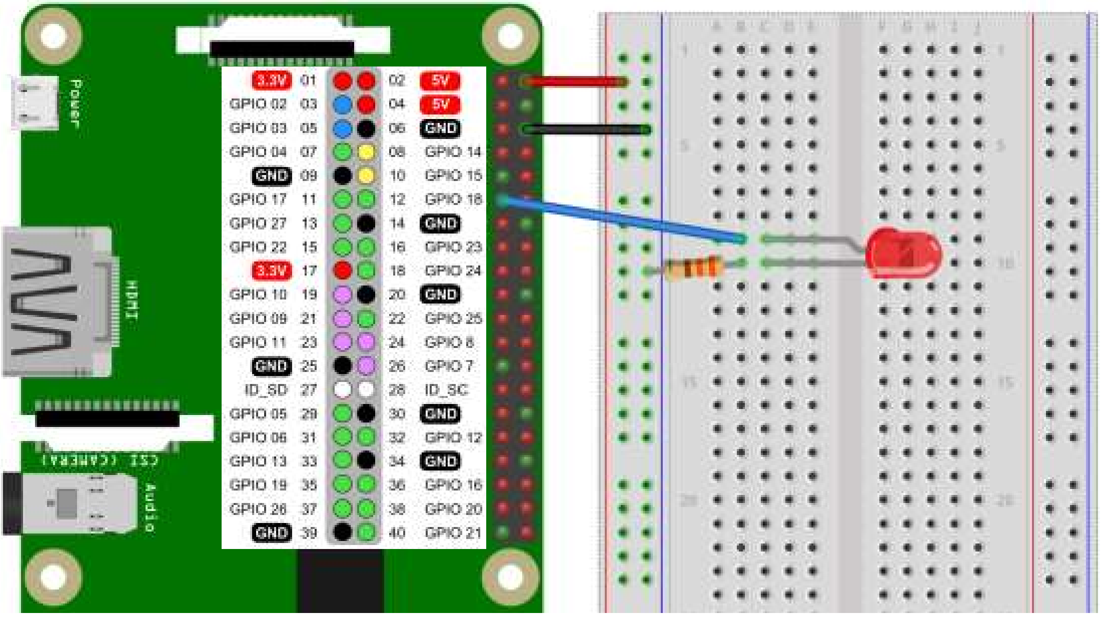


Publisher Raspberry Pi



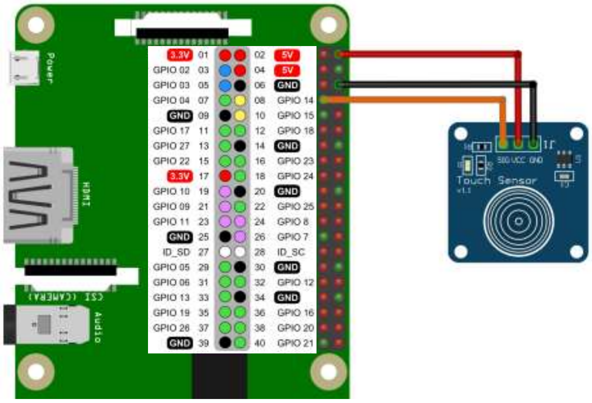
3.3V	01	02	5V
GPIO 02	03	04	5V
GPIO 03	05	06	GND
GPIO 04	07	08	GPIO 14
GND	09	10	GPIO 15
GPIO 17	11	12	GPIO 18
GPIO 27	13	14	GND
GPIO 22	15	16	GPIO 23
3.3V	17	18	GPIO 24
GPIO 10	19	20	GND
GPIO 09	21	22	GPIO 25
GPIO 11	23	24	GPIO 8
GND	25	26	GPIO 7
ID_SD	27	28	ID_SC
GPIO 05	29	30	GND
GPIO 06	31	32	GPIO 12
GPIO 13	33	34	GND
GPIO 19	35	36	GPIO 16
GPIO 26	37	38	GPIO 20
GND	39	40	GPIO 21

Subscriber Raspberry Pi

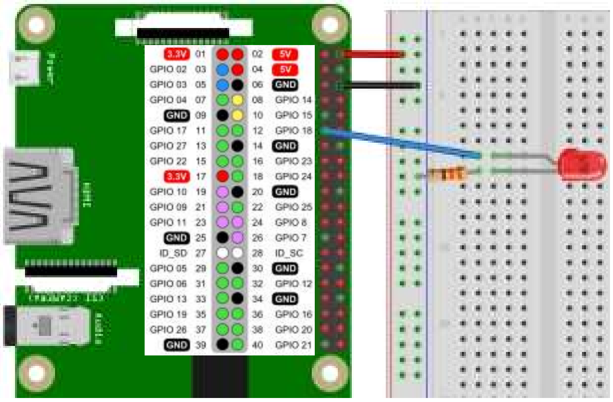


3.3V	01	02	5V
GPIO 02	03	04	5V
GPIO 03	05	06	GND
GPIO 04	07	08	GPIO 14
GND	09	10	GPIO 15
GPIO 17	11	12	GPIO 18
GPIO 27	13	14	GND
GPIO 22	15	16	GPIO 23
3.3V	17	18	GPIO 24
GPIO 10	19	20	GND
GPIO 09	21	22	GPIO 25
GPIO 11	23	24	GPIO 8
GND	25	26	GPIO 7
ID_SD	27	28	ID_SC
GPIO 05	29	30	GND
GPIO 06	31	32	GPIO 12
GPIO 13	33	34	GND
GPIO 19	35	36	GPIO 16
GPIO 26	37	38	GPIO 20
GND	39	40	GPIO 21

Publisher and Subscriber

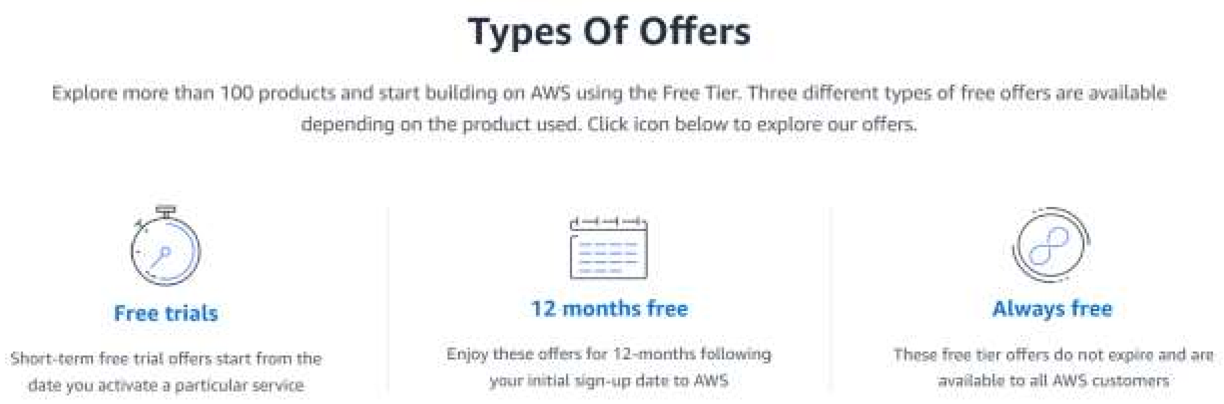


Publisher RPi



Subscriber RPi

AWS: Free Tier



Types Of Offers

Explore more than 100 products and start building on AWS using the Free Tier. Three different types of free offers are available depending on the product used. Click icon below to explore our offers.

Free trials

Short-term free trial offers start from the date you activate a particular service

12 months free

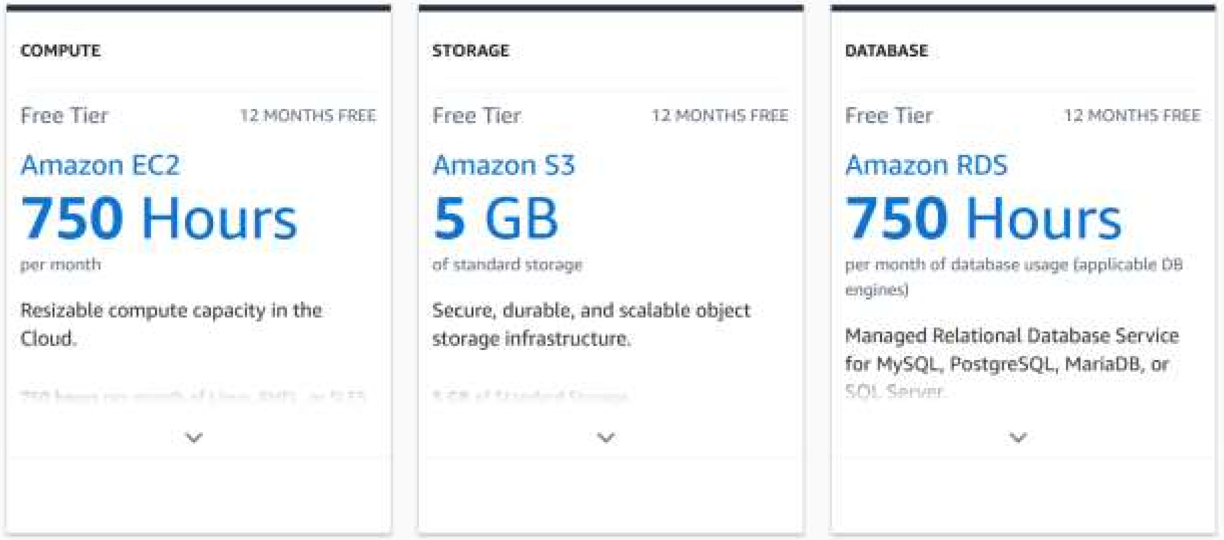
Enjoy these offers for 12-months following your initial sign-up date to AWS

Always free

These free tier offers do not expire and are available to all AWS customers

https://aws.amazon.com/free

AWS: Products and Services



COMPUTE

Free Tier 12 MONTHS FREE

Amazon EC2

750 Hours

per month

Resizable compute capacity in the Cloud.

750 hours per month of Linux, RHEL, or SLES

STORAGE

Free Tier 12 MONTHS FREE

Amazon S3

5 GB

of standard storage

Secure, durable, and scalable object storage infrastructure.

5 GB of Standard Storage

DATABASE

Free Tier 12 MONTHS FREE

Amazon RDS

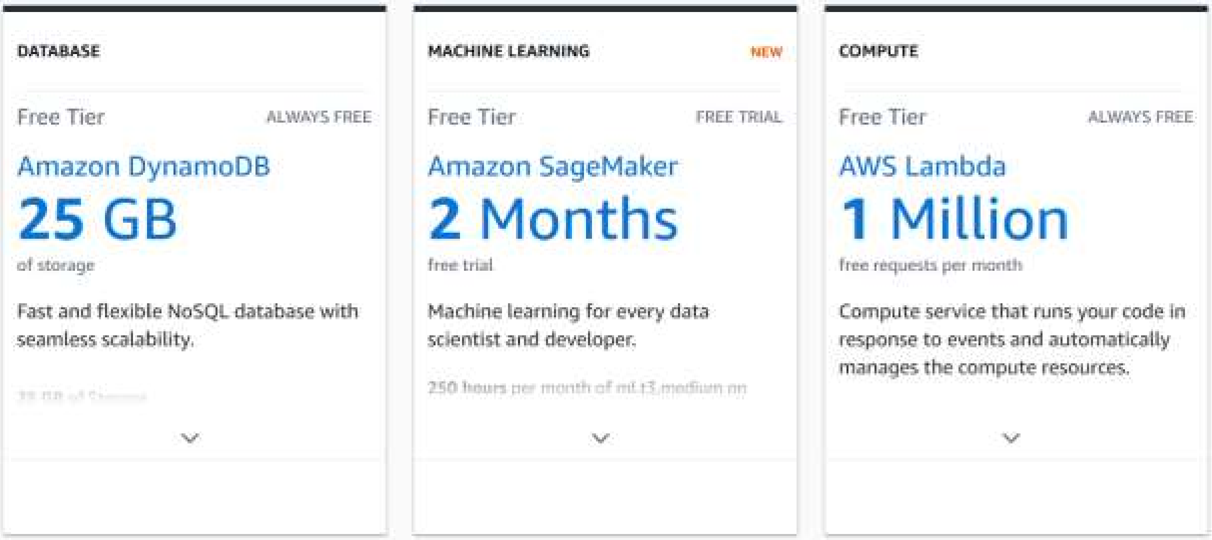
750 Hours

per month of database usage (applicable DB engines)

Managed Relational Database Service for MySQL, PostgreSQL, MariaDB, or SQL Server.

https://aws.amazon.com/free

AWS: Products and Services



DATABASE

Free Tier ALWAYS FREE

Amazon DynamoDB

25 GB

of storage

Fast and flexible NoSQL database with seamless scalability.

25 GB of Storage

MACHINE LEARNING NEW

Free Tier FREE TRIAL

Amazon SageMaker

2 Months

free trial

Machine learning for every data scientist and developer.

250 hours per month of ml.t3.medium on

COMPUTE

Free Tier ALWAYS FREE

AWS Lambda

1 Million

free requests per month

Compute service that runs your code in response to events and automatically manages the compute resources.

https://aws.amazon.com/free

21

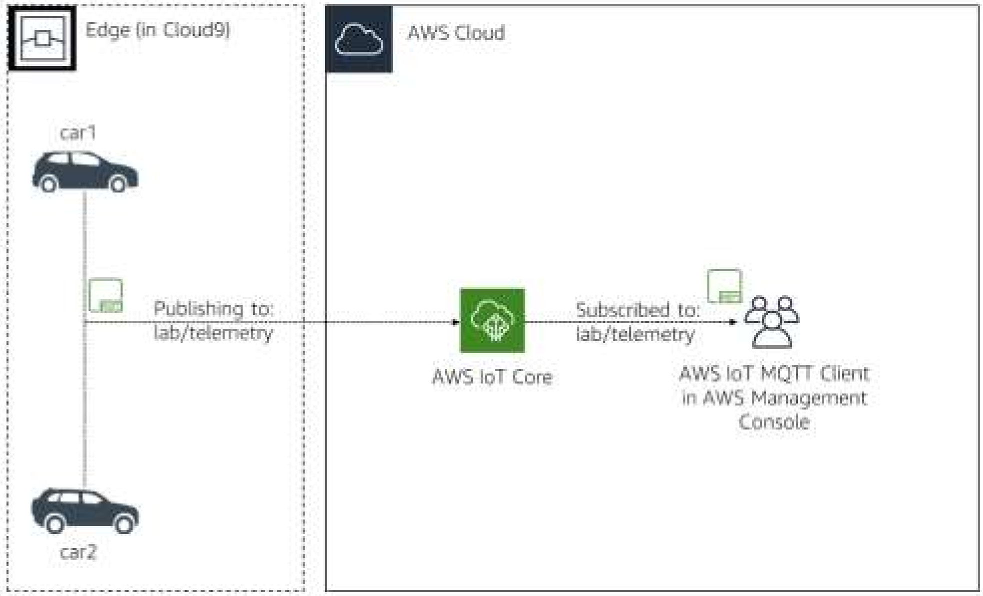
AWS: Products and Services

- **Internet of Things**
- Machine Learning
- Robotics
- Database
- Front-End Web & Mobile
- Analytics
- Security, Identity & Compliance
- Application Integration
- Business Productivity
- Developer Tools
- Networking

https://aws.amazon.com/free

22

IoT Application



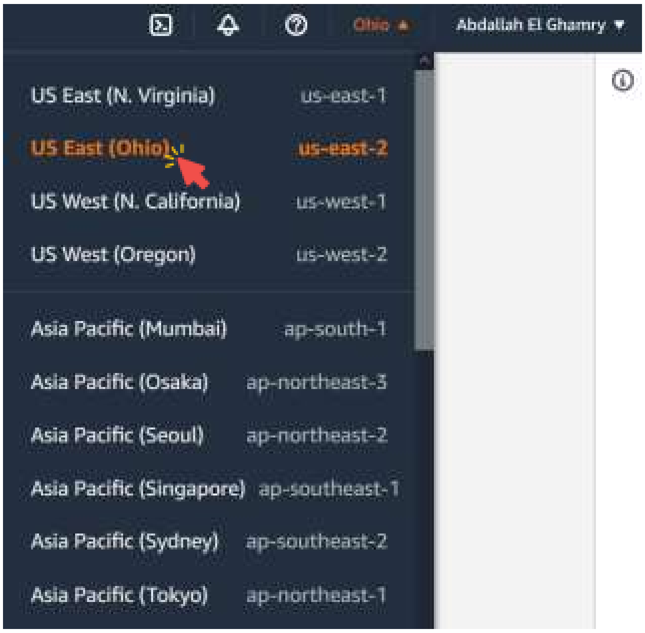
Create an AWS Cloud9 Environment

1. Make sure you are in the Frankfurt, Ireland, N. Virginia, Ohio, Oregon or Tokyo Region.

   Those are, at the time of writing the exercises, the only regions with all the services that will be used: Amazon EC2, Amazon S3, Amazon SNS, AWS Cloud9, AWS IAM, AWS IoT Analytics, AWS IoT Core, AWS IoT Greengrass, AWS Lambda.

   Since all resources must be in the same region for the exercises to work, you must use a region where all those services are available.

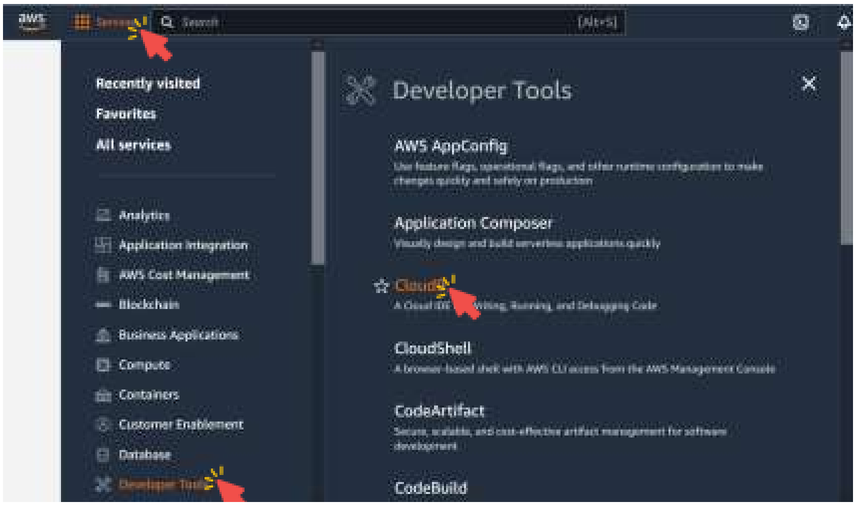
Create an AWS Cloud9 Environment



US East (N. Virginia)	us-east-1
US East (Ohio)	us-east-2
US West (N. California)	us-west-1
US West (Oregon)	us-west-2
Asia Pacific (Mumbai)	ap-south-1
Asia Pacific (Osaka)	ap-northeast-3
Asia Pacific (Seoul)	ap-northeast-2
Asia Pacific (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Asia Pacific (Tokyo)	ap-northeast-1

Create an AWS Cloud9 Environment

2. In the AWS Management Console, click Services, click Developer Tools, and then click Cloud9 to open the Cloud9 dashboard.



AWS AppConfig
Use feature flags, operational flags, and other runtime configuration to make changes quickly and safely on production

Application Composer
Visually design and build serverless applications quickly

☆ Cloud9
A Cloud IDE Writing, Running, and Debugging Code

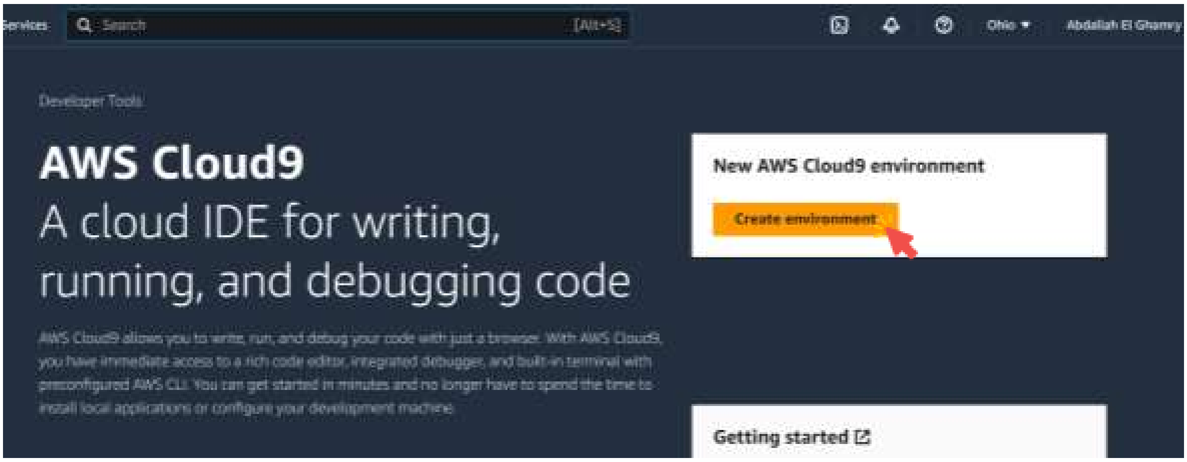
CloudShell
A browser-based shell with AWS CLI access from the AWS Management Console

CodeArtifact
Secure, scalable, and cost-effective artifact management for software development

CodeBuild

Create an AWS Cloud9 Environment

3. Click Create environment at the top-right corner.



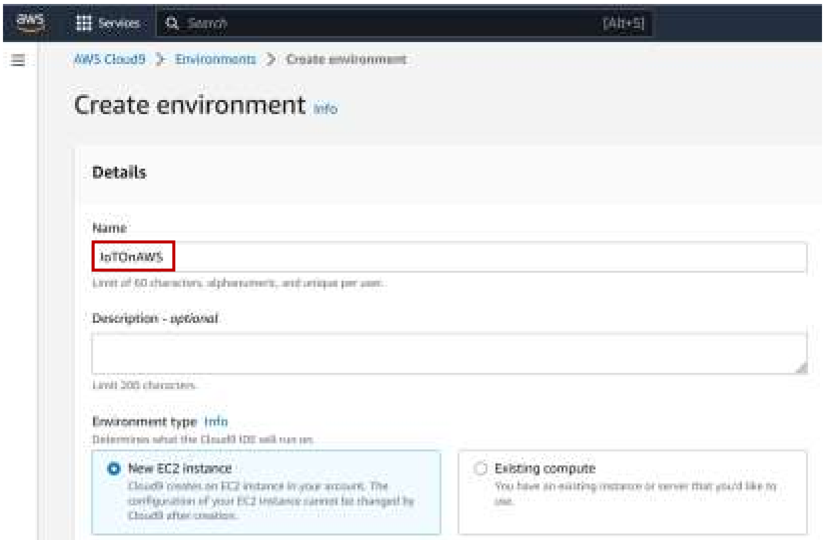
AWS Cloud9

A cloud IDE for writing, running, and debugging code

AWS Cloud9 allows you to write, run, and debug your code with just a browser. With AWS Cloud9, you have immediate access to a rich code editor, integrated debugger, and built-in terminal with preconfigured AWS CLI. You can get started in minutes and no longer have to spend the time to install local applications or configure your development machine.

Create an AWS Cloud9 Environment

4. For Name, enter `IoTOnAWS`.



Create environment

Details

Name: IoTOnAWS

Limit of 60 characters, alphanumeric, and unique per user.

Description - *optional*

Limit 200 characters.

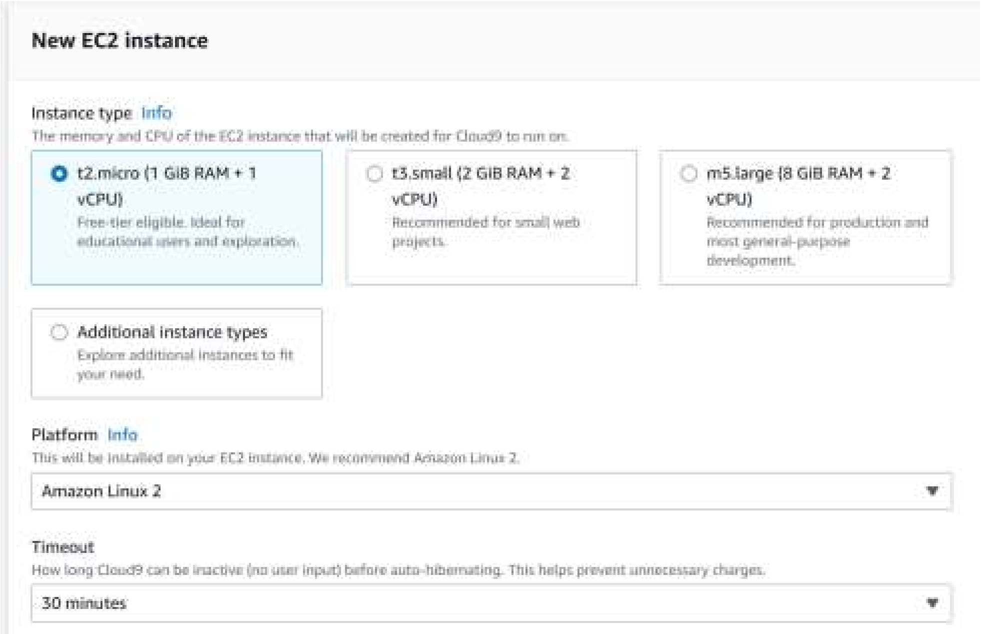
Environment type

Determines what the Cloud9 IDE will run on.

● New EC2 instance
Cloud9 creates an EC2 instance in your account. The configuration of your EC2 instance cannot be changed by Cloud9 after creation.

○ Existing compute
You have an existing instance or server that you'd like to use.

Create an AWS Cloud9 Environment



New EC2 instance

Instance type Info

The memory and CPU of the EC2 instance that will be created for Cloud9 to run on.

- ● t2.micro (1 GiB RAM + 1 vCPU)
  Free-tier eligible. Ideal for educational users and exploration.
- ○ t3.small (2 GiB RAM + 2 vCPU)
  Recommended for small web projects.
- ○ m5.large (8 GiB RAM + 2 vCPU)
  Recommended for production and most general-purpose development.
- ○ Additional instance types
  Explore additional instances to fit your need.

Platform Info

This will be installed on your EC2 instance. We recommend Amazon Linux 2.

Amazon Linux 2 ▼

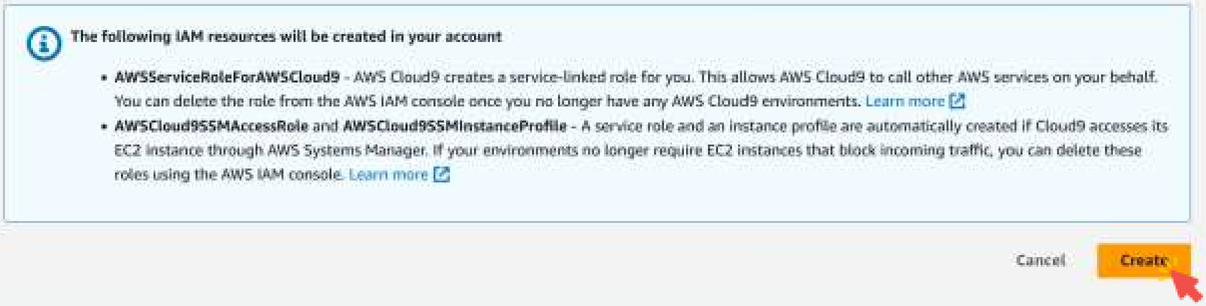
Timeout

How long Cloud9 can be inactive (no user input) before auto-hibernating. This helps prevent unnecessary charges.

30 minutes ▼

Create an AWS Cloud9 Environment

5. Click Create.

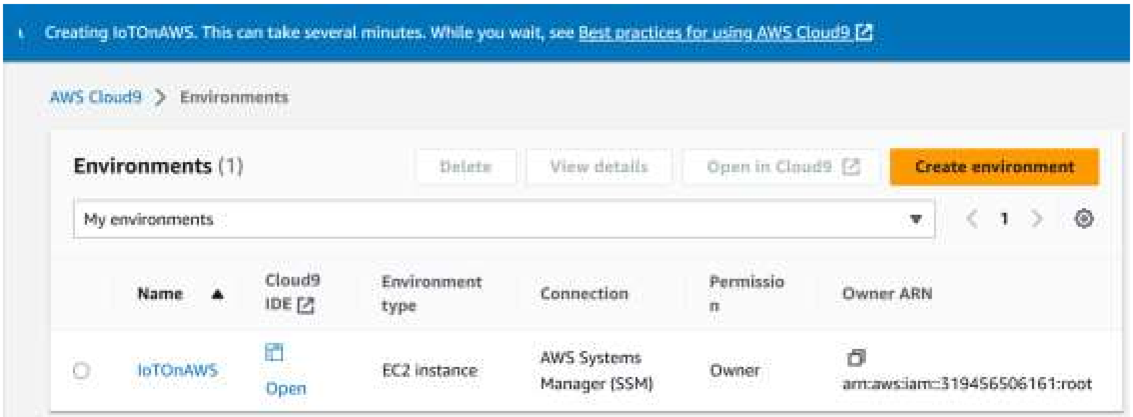


The following IAM resources will be created in your account

- **AWSServiceRoleForAWSCloud9** - AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments. Learn more
- **AWSCloud9SSMAccessRole** and **AWSCloud9SSMInstanceProfile** - A service role and an instance profile are automatically created if Cloud9 accesses its EC2 instance through AWS Systems Manager. If your environments no longer require EC2 instances that block incoming traffic, you can delete these roles using the AWS IAM console. Learn more

Create an AWS Cloud9 Environment

This should launch your AWS Cloud9 environment within a few minutes.



Creating IoTOnAWS. This can take several minutes. While you wait, see Best practices for using AWS Cloud9

AWS Cloud9 > Environments

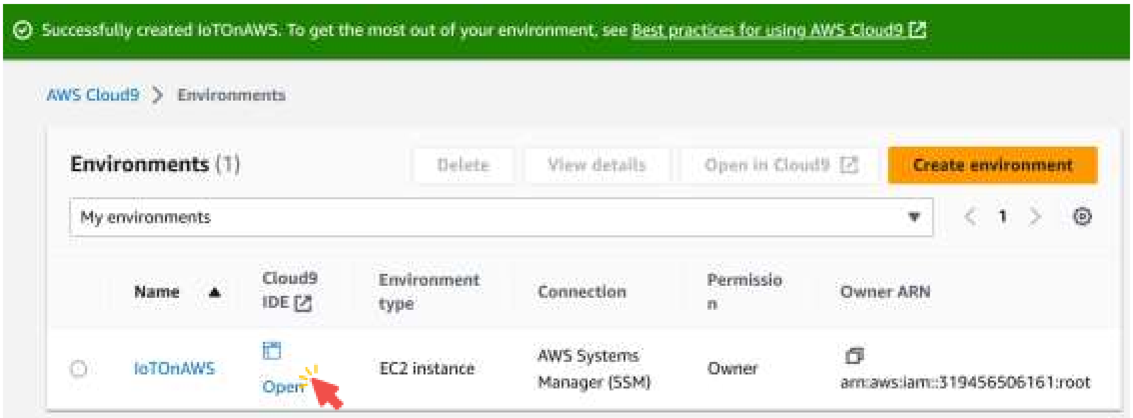
Environments (1)

	Name ▲	Cloud9 IDE	Environment type	Connection	Permission	Owner ARN
○	IoTOnAWS	Open	EC2 instance	AWS Systems Manager (SSM)	Owner	arn:aws:iam::319456506161:root

31

Create an AWS Cloud9 Environment

6. Click Open to launch your environment.



Successfully created IoTOnAWS. To get the most out of your environment, see Best practices for using AWS Cloud9

AWS Cloud9 > Environments

Environments (1)

	Name ▲	Cloud9 IDE	Environment type	Connection	Permission	Owner ARN
○	IoTOnAWS	Open	EC2 instance	AWS Systems Manager (SSM)	Owner	arn:aws:iam::319456506161:root

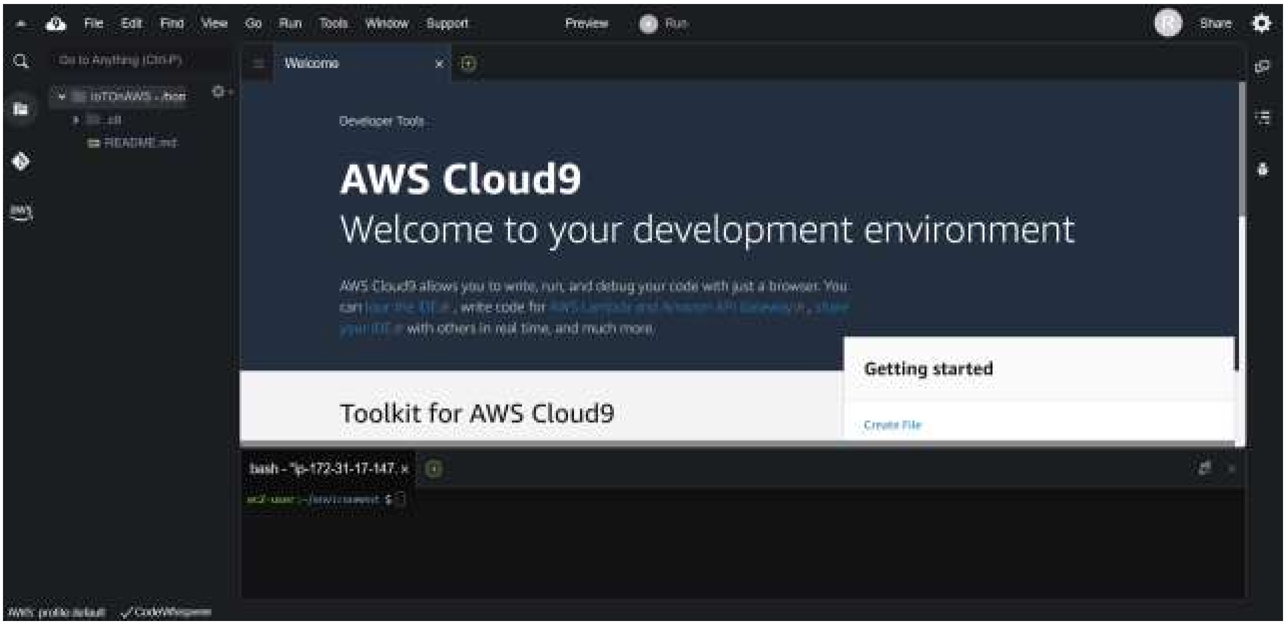
32

Create an AWS Cloud9 Environment

Press Alt-` to go to the next IDE tab.

33

Create an AWS Cloud9 Environment



AWS Cloud9

Welcome to your development environment

AWS Cloud9 allows you to write, run, and debug your code with just a browser. You can tour the IDE, write code for AWS Lambda and Amazon API Gateway, share your IDE with others in real time, and much more.

Toolkit for AWS Cloud9

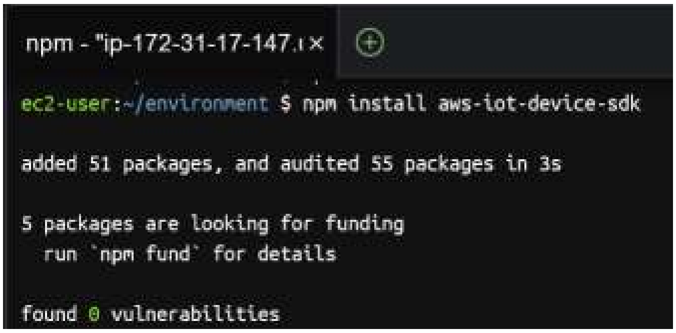
```
ec2-user:~/environment $
```

34

Download The Car Code

1. Install the AWS IoT Device SDK Node package by running the following command in your AWS Cloud9 terminal.

```
npm install aws-iot-device-sdk
```

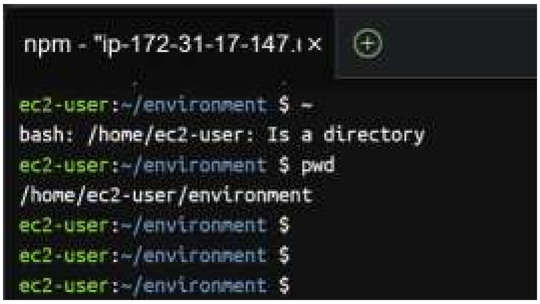


```
ec2-user:~/environment $ npm install aws-iot-device-sdk

added 51 packages, and audited 55 packages in 3s

5 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
```

35

Download The Car Code

- Remember, ~ is best known for expanding to a user's home directory.
- The pwd command (print working directory) writes the full pathname of the current working directory.



```
ec2-user:~/environment $ ~
bash: /home/ec2-user: Is a directory
ec2-user:~/environment $ pwd
/home/ec2-user/environment
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
```

36

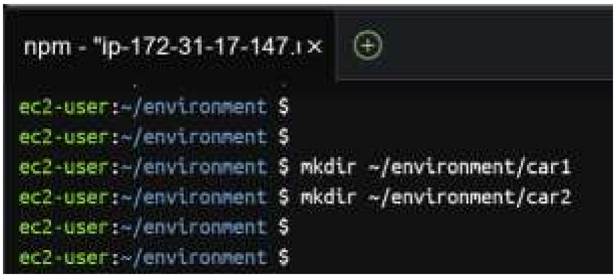
Download The Car Code

2. Create the repository structure for the Car application.

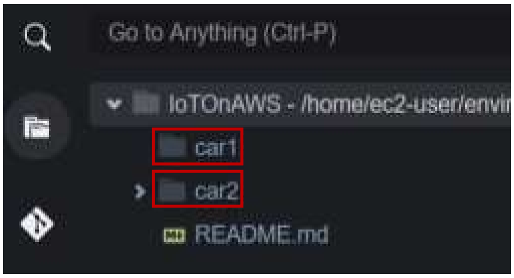
   As there will be 2 Cars, you will create 2 folders.

   Run the following commands in your AWS Cloud9 terminal.

```
mkdir ~/environment/car1
mkdir ~/environment/car2
```



```
npm - "ip-172-31-17-147.i ×
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $ mkdir ~/environment/car1
ec2-user:~/environment $ mkdir ~/environment/car2
ec2-user:~/environment $
ec2-user:~/environment $
```



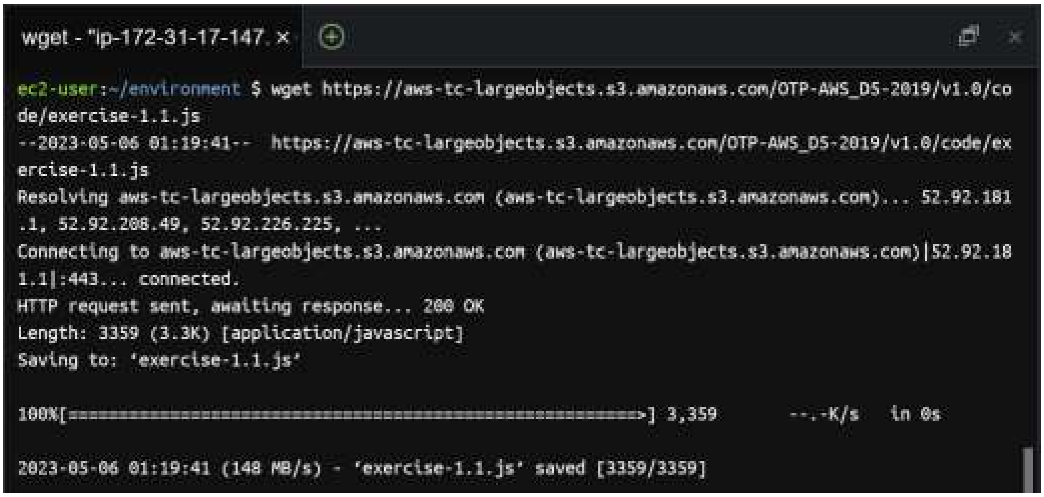
Download The Car Code

3. Download and copy the application code in each car folder by running the following commands in your AWS Cloud9 terminal.

```
cd ~/environment
wget  https://aws-tc-largeobjects.s3.amazonaws.com/OTP-
AWS_D5-2019/v1.0/code/exercise-1.1.js
cp exercise-1.1.js car1
cp exercise-1.1.js car2
rm exercise-1.1.js
```

Download The Car Code

```
wget  https://aws-tc-largeobjects.s3.amazonaws.com/OTP-AWS_D5-2019/v1.0/code/exercise-1.1.js
```

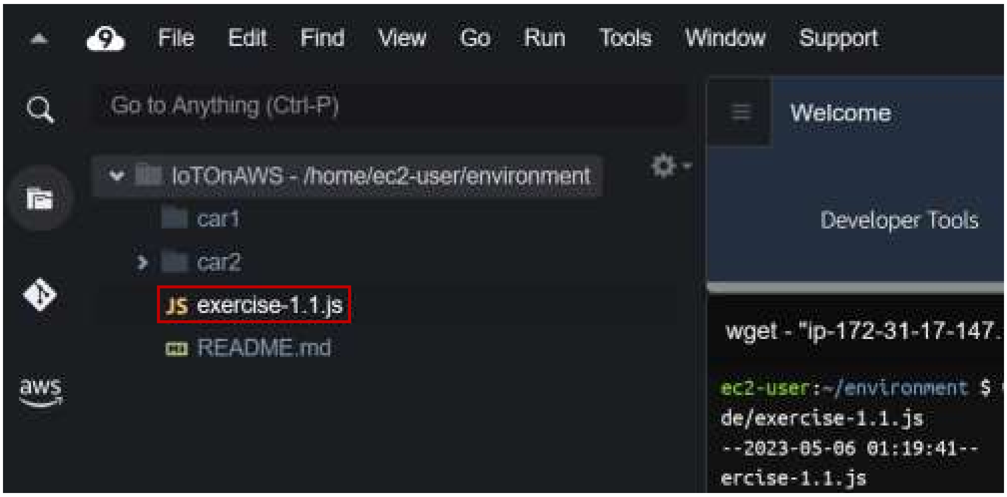


```
ec2-user:~/environment $ wget https://aws-tc-largeobjects.s3.amazonaws.com/OTP-AWS_D5-2019/v1.0/co
de/exercise-1.1.js
--2023-05-06 01:19:41--  https://aws-tc-largeobjects.s3.amazonaws.com/OTP-AWS_D5-2019/v1.0/code/ex
ercise-1.1.js
Resolving aws-tc-largeobjects.s3.amazonaws.com (aws-tc-largeobjects.s3.amazonaws.com)... 52.92.181
.1, 52.92.208.49, 52.92.226.225, ...
Connecting to aws-tc-largeobjects.s3.amazonaws.com (aws-tc-largeobjects.s3.amazonaws.com)|52.92.18
1.1|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 3359 (3.3K) [application/javascript]
Saving to: ‘exercise-1.1.js’

100%[=============================================================>] 3,359       --.-K/s   in 0s

2023-05-06 01:19:41 (148 MB/s) - ‘exercise-1.1.js’ saved [3359/3359]
```

39

Download The Car Code

```
wget  https://aws-tc-largeobjects.s3.amazonaws.com/OTP-AWS_D5-2019/v1.0/code/exercise-1.1.js
```

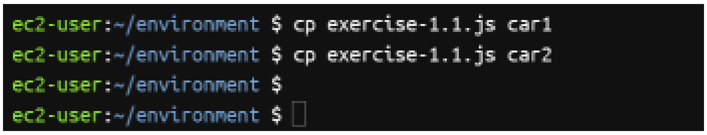


```
ec2-user:~/environment $
de/exercise-1.1.js
--2023-05-06 01:19:41--
ercise-1.1.js
```

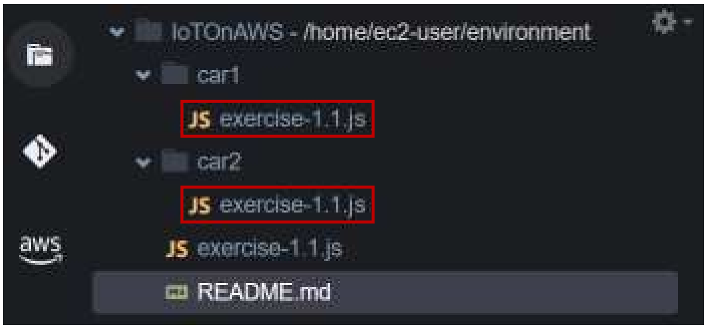
40

Download The Car Code

```
cp exercise-1.1.js car1
cp exercise-1.1.js car2
```

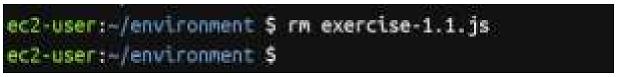


```
ec2-user:~/environment $ cp exercise-1.1.js car1
ec2-user:~/environment $ cp exercise-1.1.js car2
ec2-user:~/environment $
ec2-user:~/environment $ 
```

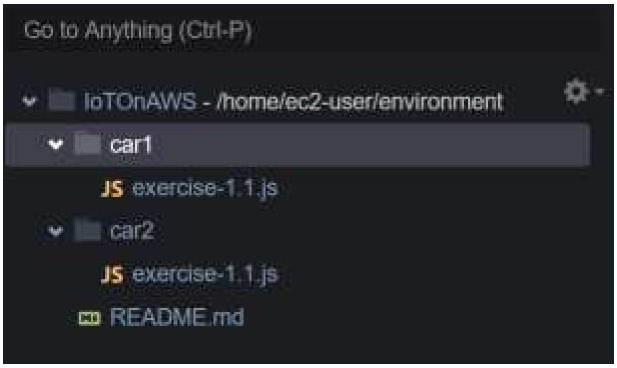


Download The Car Code

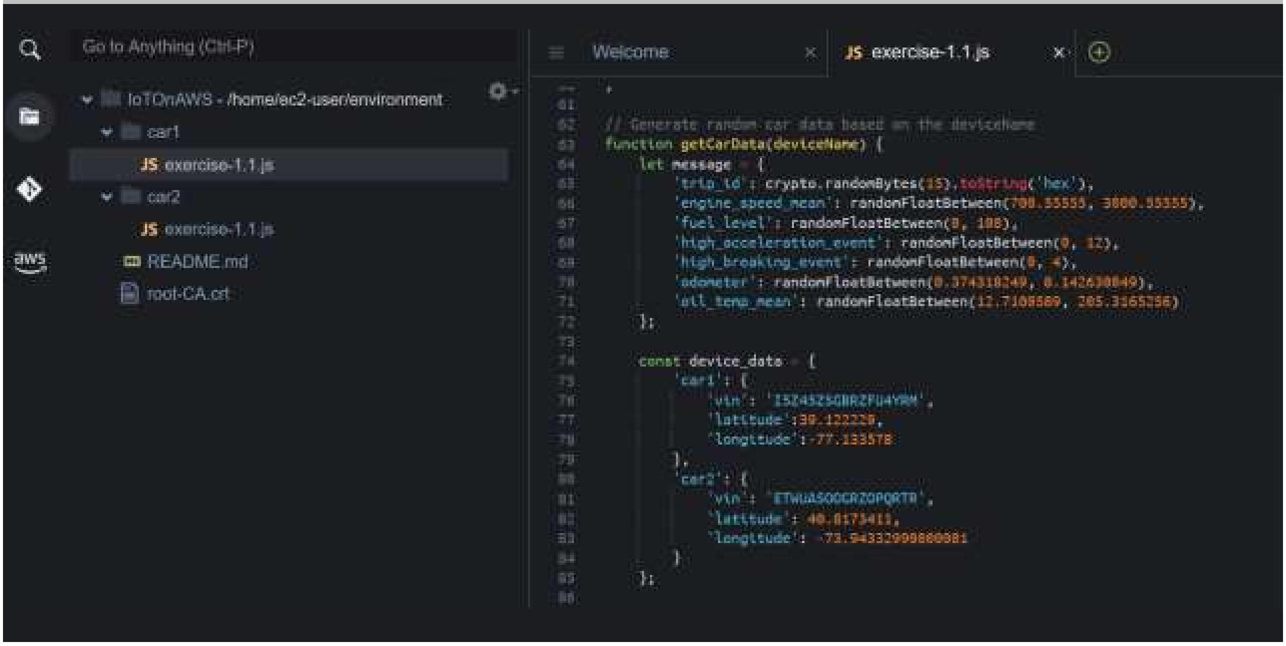
```
rm exercise-1.1.js
```



```
ec2-user:~/environment $ rm exercise-1.1.js
ec2-user:~/environment $
```



Download The Car Code



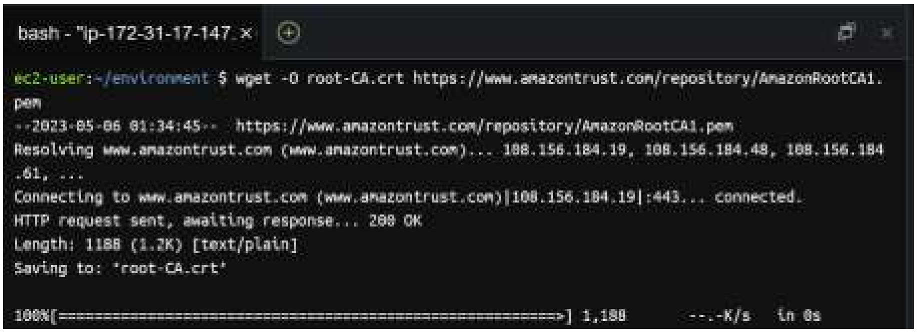
```
// Generate random car data based on the deviceName
function getCarData(deviceName) {
    let message = {
        'trip_id': crypto.randomBytes(15).toString('hex'),
        'engine_speed_mean': randomFloatBetween(700.55555, 3000.55555),
        'fuel_level': randomFloatBetween(0, 100),
        'high_acceleration_event': randomFloatBetween(0, 12),
        'high_breaking_event': randomFloatBetween(0, 4),
        'odometer': randomFloatBetween(0.374318249, 0.142630049),
        'oil_temp_mean': randomFloatBetween(12.7100589, 205.3165256)
    };

    const device_data = {
        'car1': {
            'vin': '1SZ4SZ5GBRZFU4YRM',
            'latitude':39.122229,
            'longitude':-77.133578
        },
        'car2': {
            'vin': 'ETWUASOOCRZOPQRTR',
            'latitude': 40.8173411,
            'longitude': -73.94332999880081
        }
    };
```

Download The Car Code

4. Download the AWS IoT Certificate Authority Public Certificate that will be used in the code later

To do so, execute the following commands in your Cloud9 terminal.

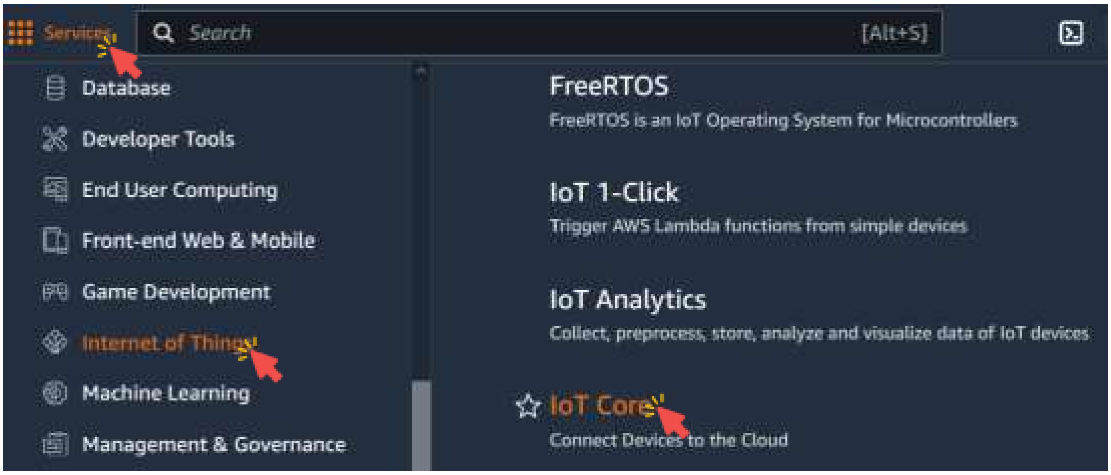
```
wget -O root-CA.crt https://www.amazontrust.com/repository/AmazonRootCA1.pem
```



```
ec2-user:~/environment $ wget -O root-CA.crt https://www.amazontrust.com/repository/AmazonRootCA1.
pem
--2023-05-06 01:34:45--  https://www.amazontrust.com/repository/AmazonRootCA1.pem
Resolving www.amazontrust.com (www.amazontrust.com)... 108.156.184.19, 108.156.184.48, 108.156.184
.61, ...
Connecting to www.amazontrust.com (www.amazontrust.com)|108.156.184.19|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1188 (1.2K) [text/plain]
Saving to: 'root-CA.crt'

100%[==========================================================>] 1,188       --.-K/s   in 0s
```

Create an IoT Thing

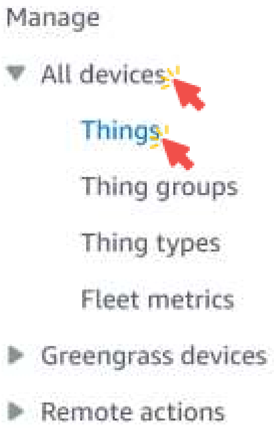
1. In the AWS Management Console, click Services, click Internet of Things and click IoT Core to open the IoT Console.


Services
Search
[Alt+S]
Database
Developer Tools
End User Computing
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
FreeRTOS
FreeRTOS is an IoT Operating System for Microcontrollers
IoT 1-Click
Trigger AWS Lambda functions from simple devices
IoT Analytics
Collect, preprocess, store, analyze and visualize data of IoT devices
IoT Core
Connect Devices to the Cloud

45

Create an IoT Thing

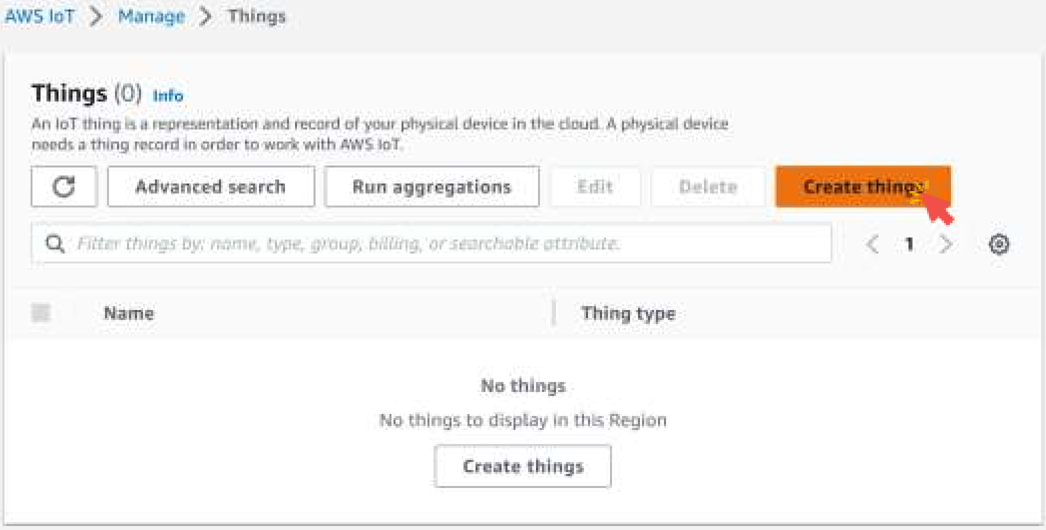
2. Expand All devices in the left menu, and click Things.



46

Create an IoT Thing

3. Click Create things.


AWS IoT > Manage > Things

Things (0) Info

An IoT thing is a representation and record of your physical device in the cloud. A physical device needs a thing record in order to work with AWS IoT.

Advanced search
Run aggregations
Edit
Delete
Create things
Filter things by: name, type, group, billing, or searchable attribute.
1
Name
Thing type

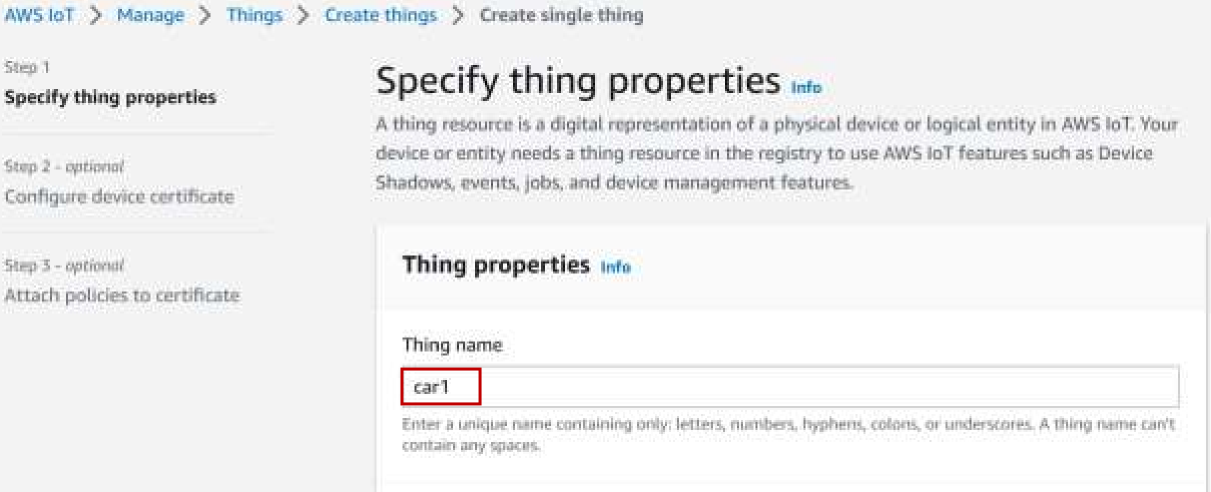
No things

No things to display in this Region

47

Create an IoT Thing

4. For Name, enter car1 and click Next.


AWS IoT > Manage > Things > Create things > Create single thing

Specify thing properties Info

A thing resource is a digital representation of a physical device or logical entity in AWS IoT. Your device or entity needs a thing resource in the registry to use AWS IoT features such as Device Shadows, events, jobs, and device management features.

Thing properties Info

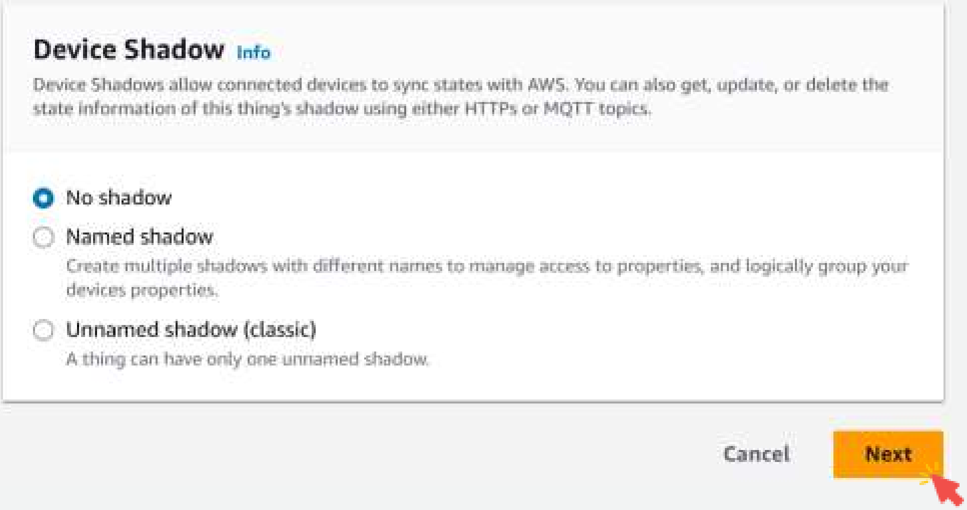
Thing name: car1

Enter a unique name containing only: letters, numbers, hyphens, colons, or underscores. A thing name can't contain any spaces.

48

Create an IoT Thing

4. For Name, enter `car1` and click Next.



Device Shadow Info

Device Shadows allow connected devices to sync states with AWS. You can also get, update, or delete the state information of this thing's shadow using either HTTPs or MQTT topics.

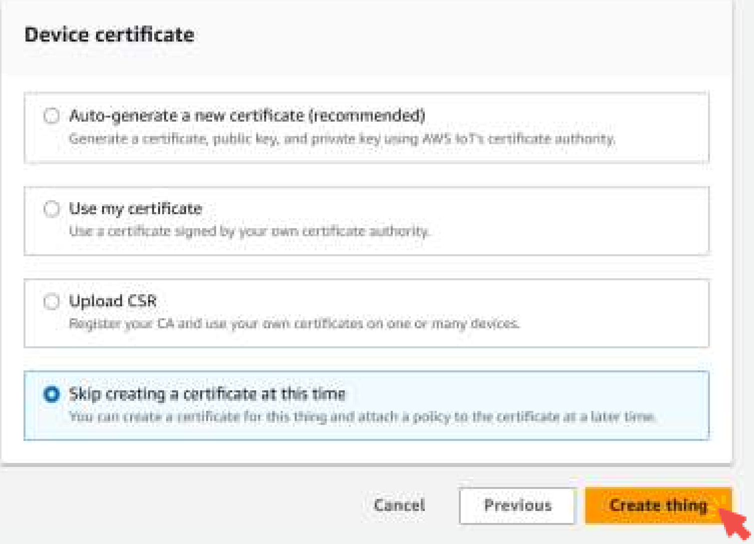
● No shadow

○ Named shadow
Create multiple shadows with different names to manage access to properties, and logically group your devices properties.

○ Unnamed shadow (classic)
A thing can have only one unnamed shadow.

Create an IoT Thing

5. Choose Skip creating a certificate at this time, and click Create thing.



Device certificate

○ Auto-generate a new certificate (recommended)
Generate a certificate, public key, and private key using AWS IoT's certificate authority.

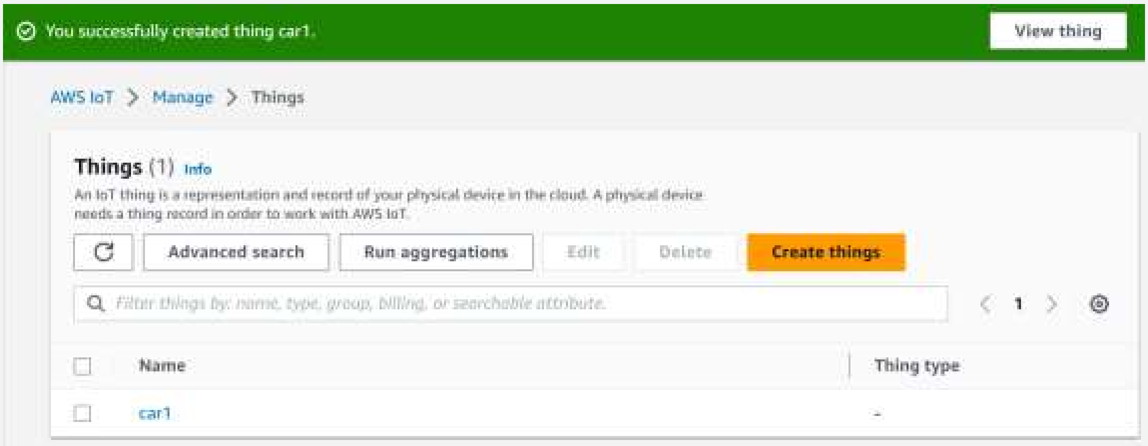
○ Use my certificate
Use a certificate signed by your own certificate authority.

○ Upload CSR
Register your CA and use your own certificates on one or many devices.

● Skip creating a certificate at this time
You can create a certificate for this thing and attach a policy to the certificate at a later time.

Create an IoT Thing

The car1 Thing has now been created.



You successfully created thing car1.

View thing
AWS IoT > Manage > Things
Things (1) Info

An IoT thing is a representation and record of your physical device in the cloud. A physical device needs a thing record in order to work with AWS IoT.

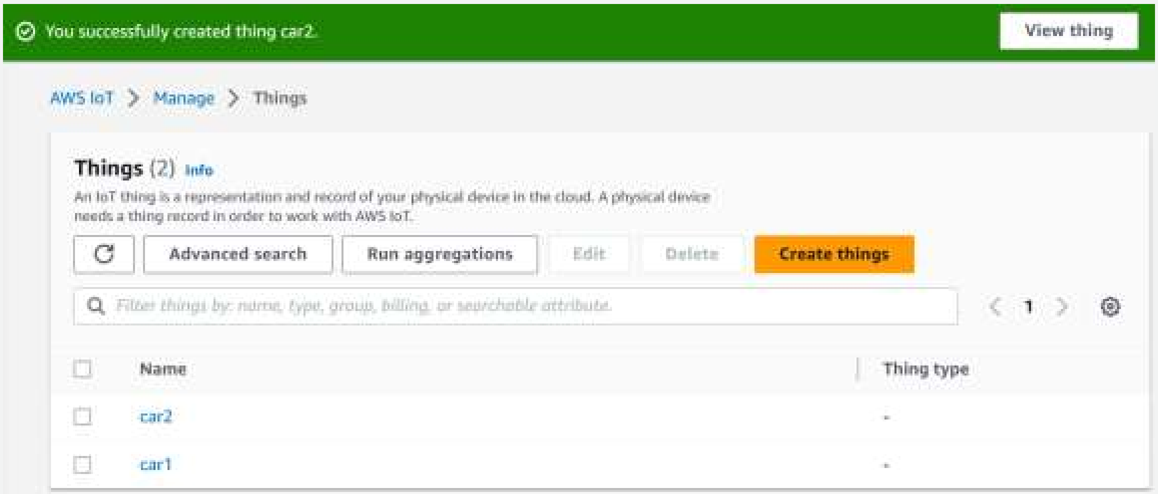
Advanced search
Run aggregations
Edit
Delete
Create things
Filter things by: name, type, group, billing, or searchable attribute.
< 1 >

□	Name	Thing type
□	car1	-

51

Create an IoT Thing

6. Repeat the same steps to create car2 thing.



You successfully created thing car2.

View thing
AWS IoT > Manage > Things
Things (2) Info

An IoT thing is a representation and record of your physical device in the cloud. A physical device needs a thing record in order to work with AWS IoT.

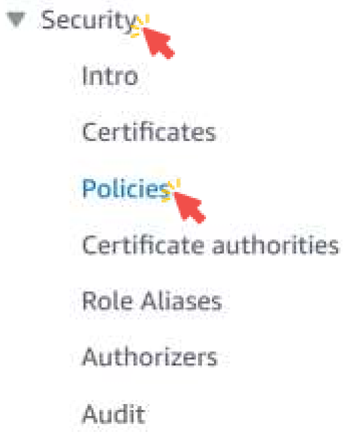
Advanced search
Run aggregations
Edit
Delete
Create things
Filter things by: name, type, group, billing, or searchable attribute.
< 1 >

□	Name	Thing type
□	car2	-
□	car1	-

52

Create an IoT Policy

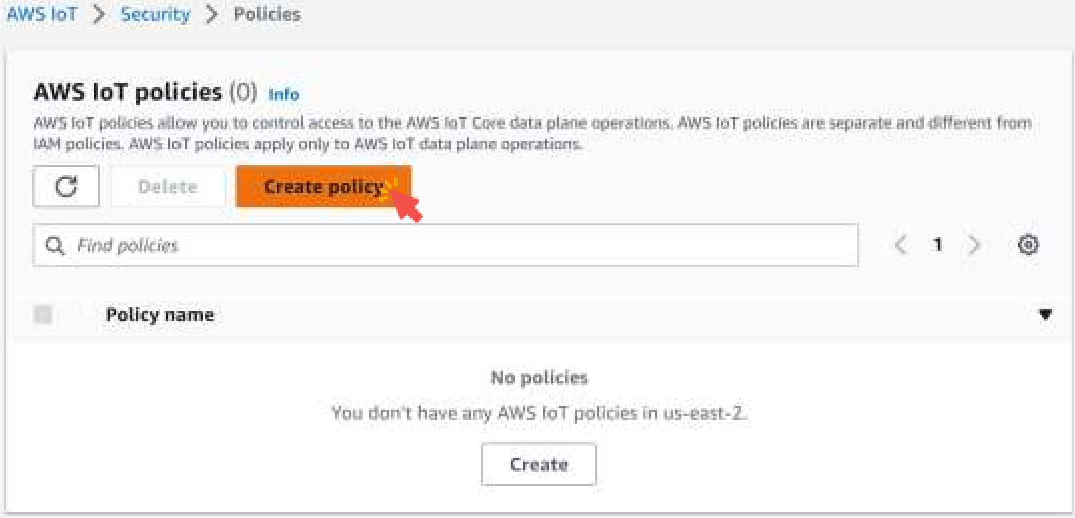
1. Expand Security in the left menu, and click Policies.



53

Create an IoT Policy

2. Click Create policy.



AWS IoT policies (0) Info

AWS IoT policies allow you to control access to the AWS IoT Core data plane operations. AWS IoT policies are separate and different from IAM policies. AWS IoT policies apply only to AWS IoT data plane operations.

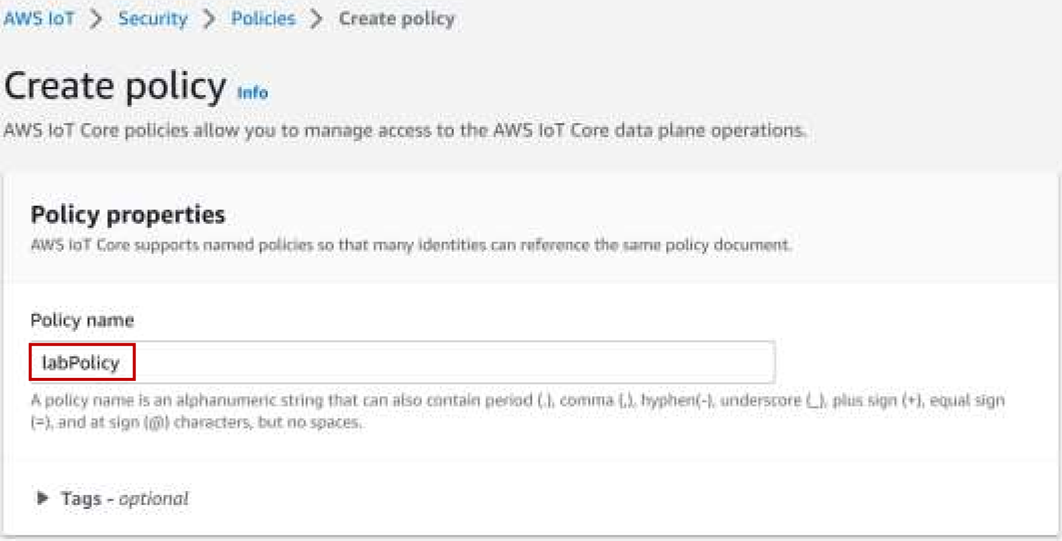
No policies

You don't have any AWS IoT policies in us-east-2.

54

Create an IoT Policy

3. For Name, enter labPolicy.



AWS IoT > Security > Policies > Create policy

Create policy Info

AWS IoT Core policies allow you to manage access to the AWS IoT Core data plane operations.

Policy properties

AWS IoT Core supports named policies so that many identities can reference the same policy document.

Policy name

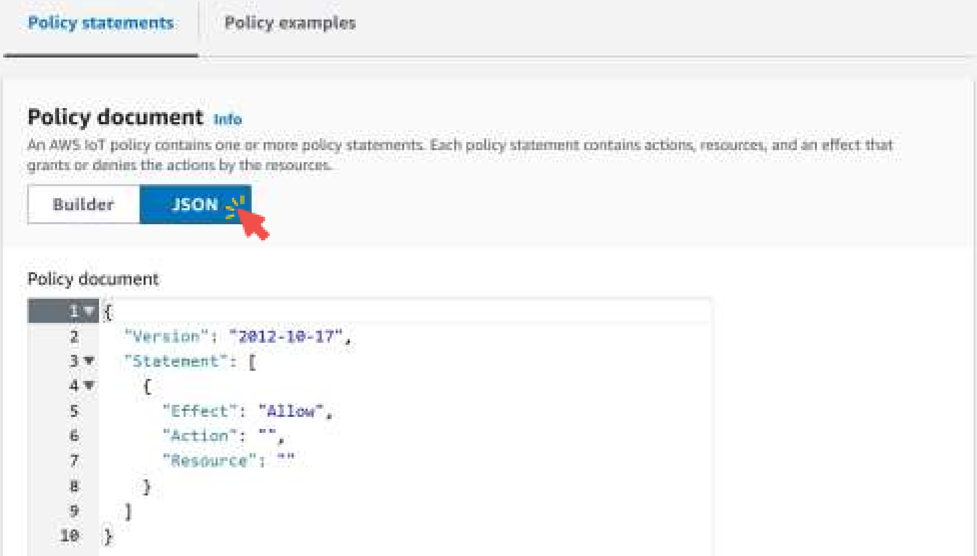
labPolicy

A policy name is an alphanumeric string that can also contain period (.), comma (,), hyphen(-), underscore (_), plus sign (+), equal sign (=), and at sign (@) characters, but no spaces.

▶ Tags - *optional*

Create an IoT Policy

4. Click on JSON.



Policy document Info

An AWS IoT policy contains one or more policy statements. Each policy statement contains actions, resources, and an effect that grants or denies the actions by the resources.

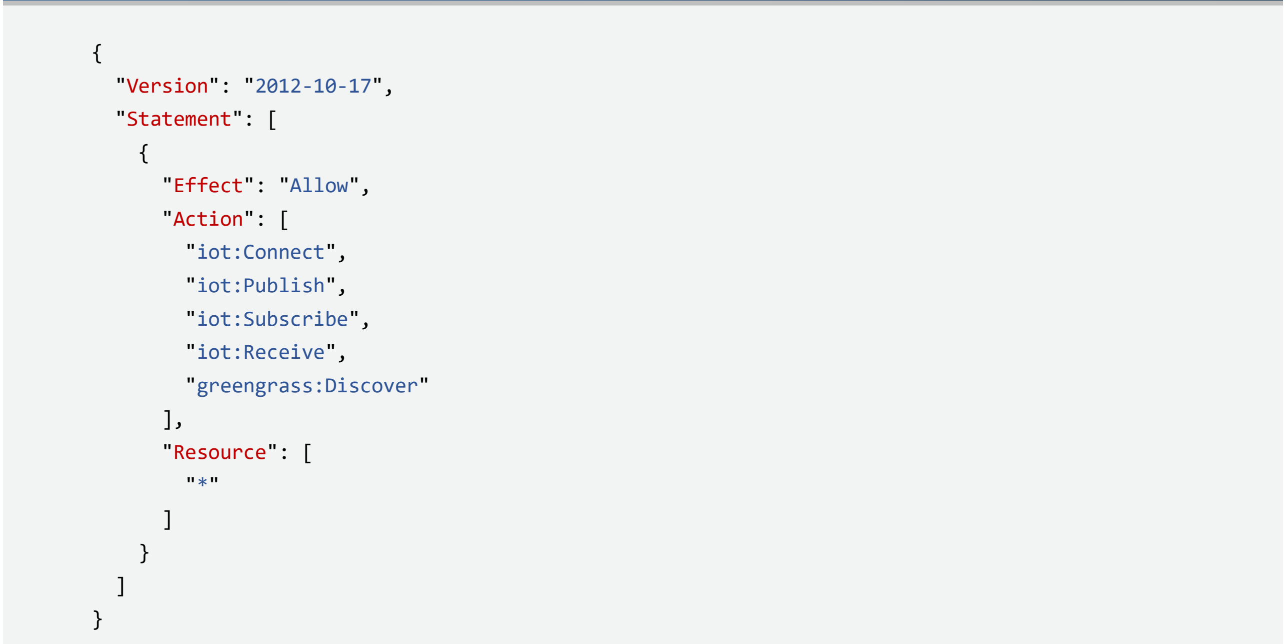
Policy document

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "",
      "Resource": ""
    }
  ]
}
```

Create an IoT Policy

5. Replace the sample policy with the following policy which authorize to Connect to your AWS IoT Core endpoint, to Publish and Subscribe to an IoT Topic, receive messages from AWS IoT once subscribed and use the Discover API from Greengrass.

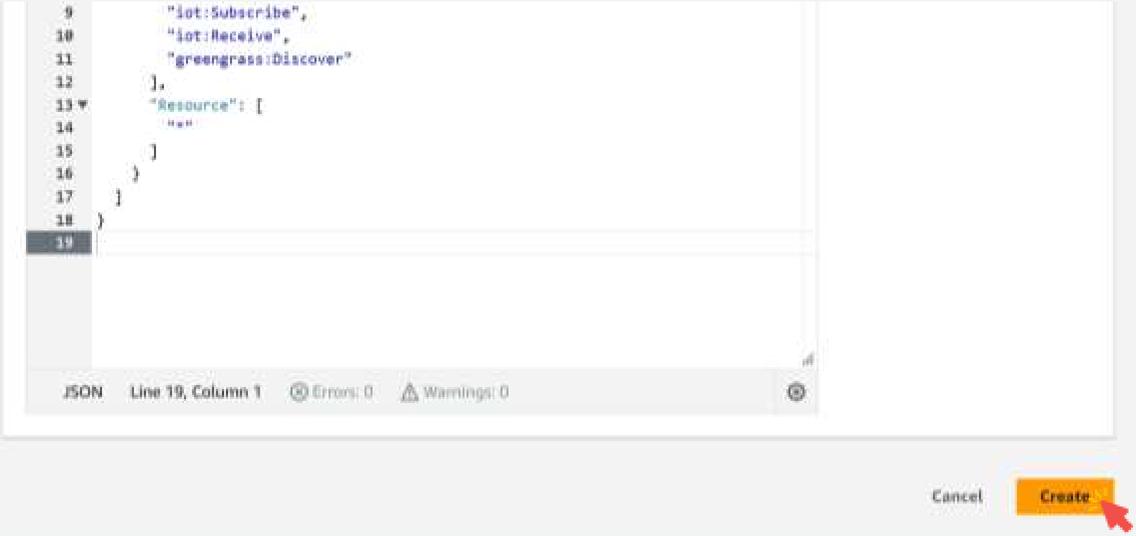
Create an IoT Policy



```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iot:Connect",
        "iot:Publish",
        "iot:Subscribe",
        "iot:Receive",
        "greengrass:Discover"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Create an IoT Policy

6. Click Create.

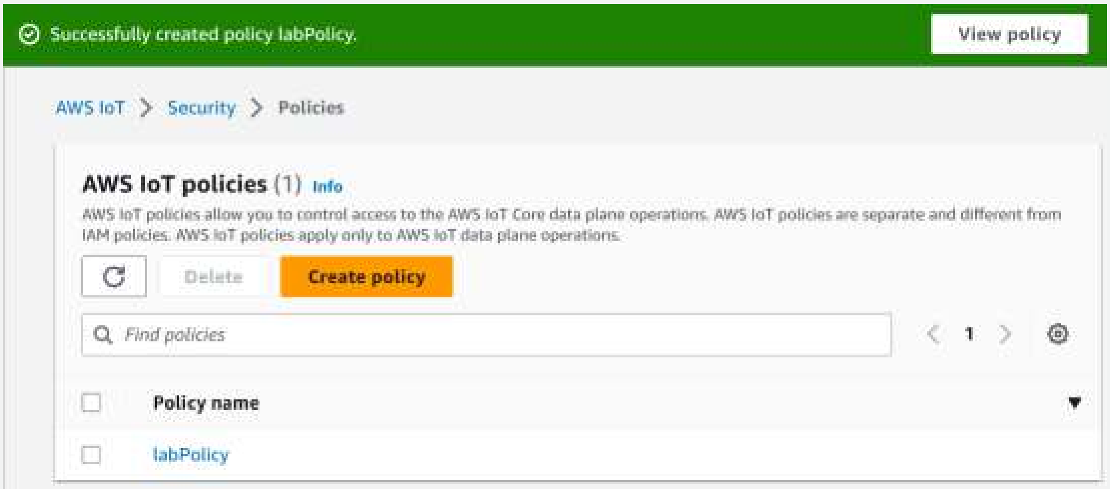


```
        "iot:Subscribe",
        "iot:Receive",
        "greengrass:Discover"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

JSON Line 19, Column 1 Errors: 0 Warnings: 0

Create an IoT Policy

You now have a Policy that provides authorizations.



Successfully created policy labPolicy.

AWS IoT > Security > Policies

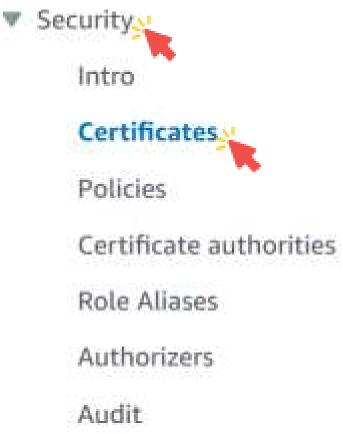
AWS IoT policies (1) Info

AWS IoT policies allow you to control access to the AWS IoT Core data plane operations. AWS IoT policies are separate and different from IAM policies. AWS IoT policies apply only to AWS IoT data plane operations.

□	Policy name
□	labPolicy

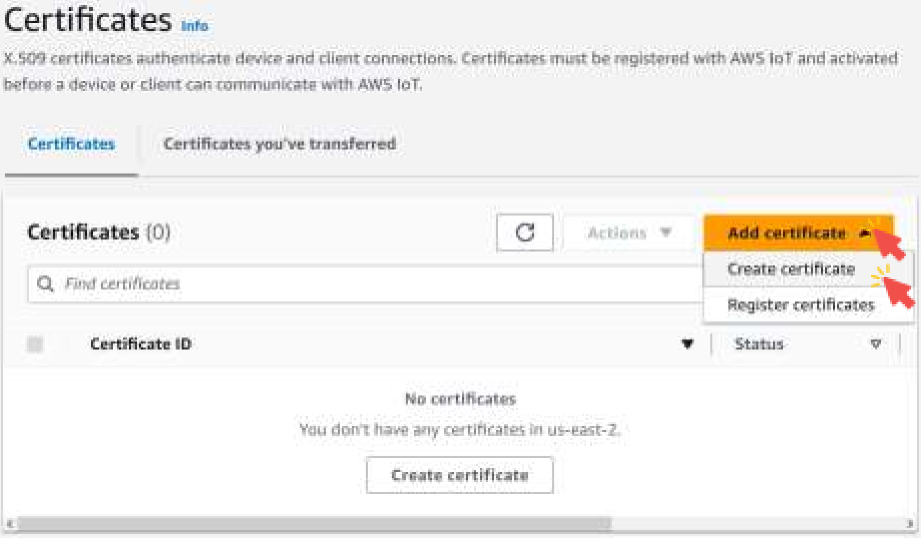
Create an IoT Certificate

1. Expand Security in the left menu, and click Certificates.



Create an IoT Certificate

2. Click Add certificate, and then click Create certificate.



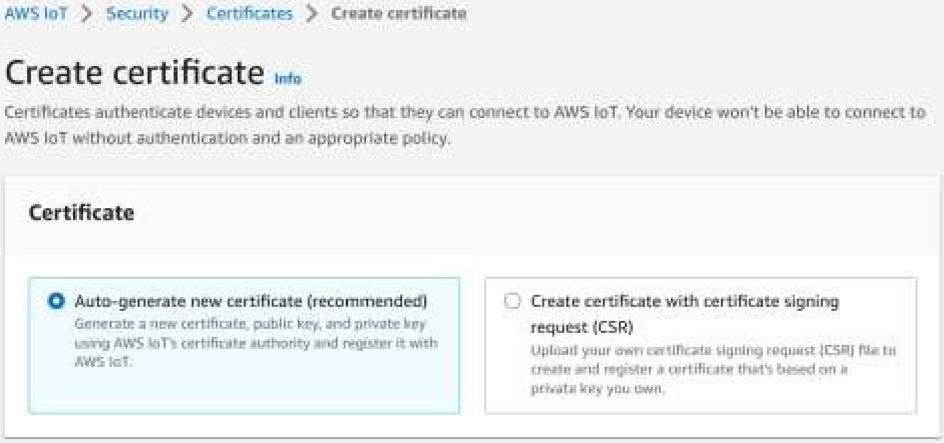
X.509 certificates authenticate device and client connections. Certificates must be registered with AWS IoT and activated before a device or client can communicate with AWS IoT.

You don't have any certificates in us-east-2.

Create an IoT Certificate

3. Choose Auto-generate new certificate.

This will automatically generate a Certificate, a Public Key and a Private Key using AWS IoT's Certificate Authority that you will then download.



AWS IoT > Security > Certificates > Create certificate

Create certificate

Certificates authenticate devices and clients so that they can connect to AWS IoT. Your device won't be able to connect to AWS IoT without authentication and an appropriate policy.

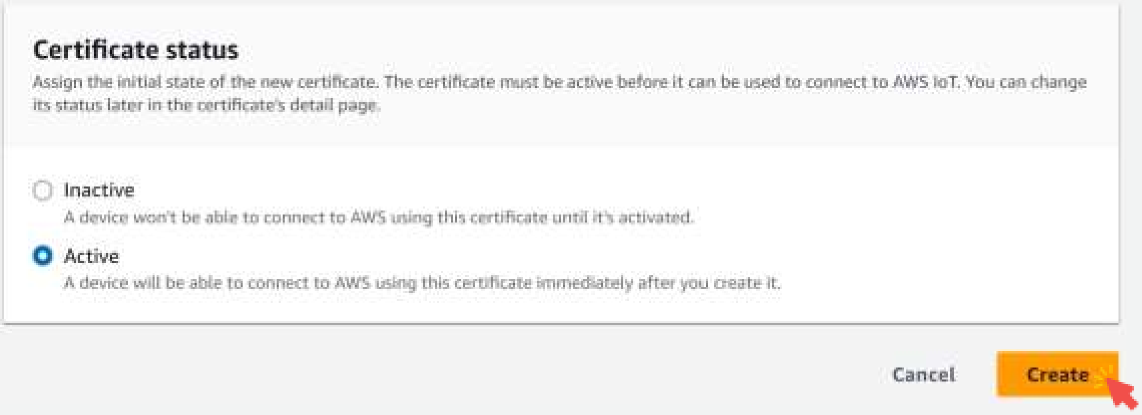
Certificate

● Auto-generate new certificate (recommended)
Generate a new certificate, public key, and private key using AWS IoT's certificate authority and register it with AWS IoT.

○ Create certificate with certificate signing request (CSR)
Upload your own certificate signing request (CSR) file to create and register a certificate that's based on a private key you own.

Create an IoT Certificate

4. Choose Active to activate the Certificate so it can be used later to connect to AWS IoT Core by your Thing, and click Create.



Certificate status

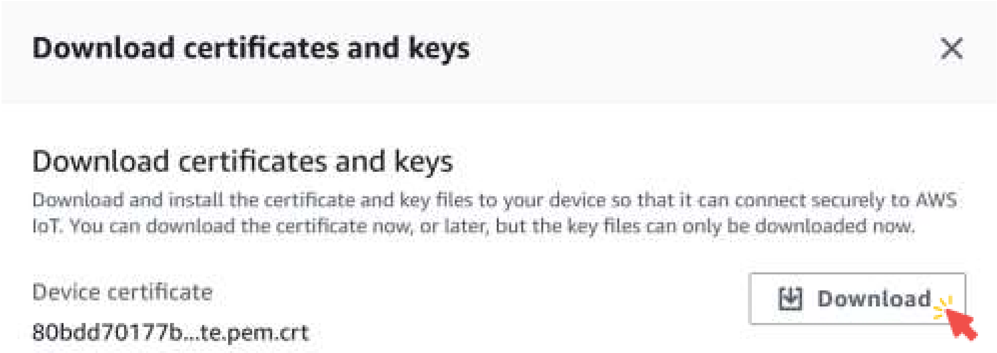
Assign the initial state of the new certificate. The certificate must be active before it can be used to connect to AWS IoT. You can change its status later in the certificate's detail page.

○ Inactive
A device won't be able to connect to AWS using this certificate until it's activated.

● Active
A device will be able to connect to AWS using this certificate immediately after you create it.

Create an IoT Certificate

5. Download the Device certificate, and rename this file to `certificate.pem.crt`.



Download certificates and keys

Download certificates and keys

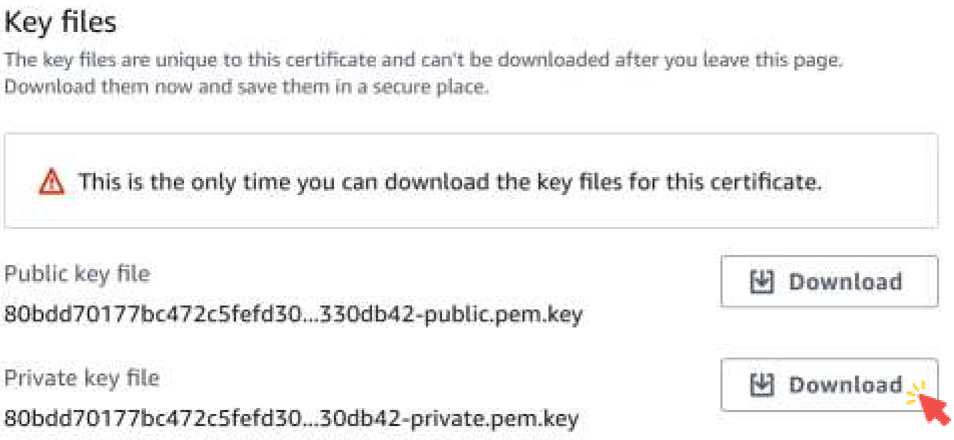
Download and install the certificate and key files to your device so that it can connect securely to AWS IoT. You can download the certificate now, or later, but the key files can only be downloaded now.

Device certificate

80bdd70177b...te.pem.crt

Create an IoT Certificate

6. Download the Private key file, and rename this file to `private.pem.key`.



Key files

The key files are unique to this certificate and can't be downloaded after you leave this page. Download them now and save them in a secure place.

⚠ This is the only time you can download the key files for this certificate.

Public key file

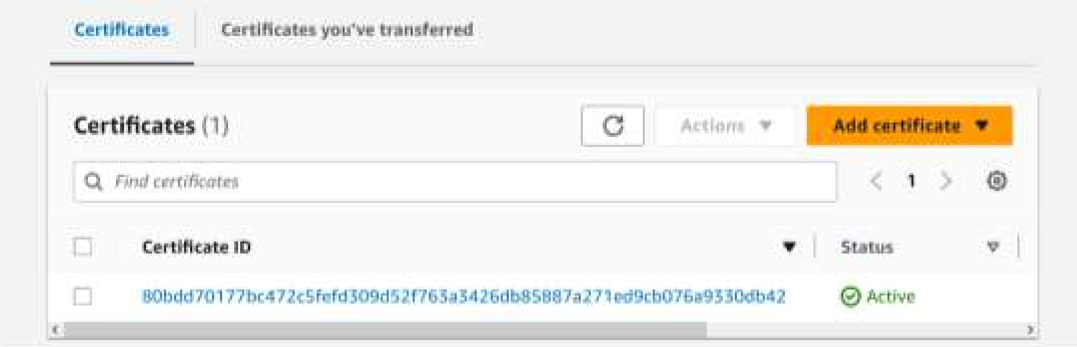
80bdd70177bc472c5fefd30...330db42-public.pem.key

Private key file

80bdd70177bc472c5fefd30...30db42-private.pem.key

Create an IoT Certificate

- You now have a Certificate and Private Key that can be used to connect to your AWS IoT Core endpoint.
- However, this is only for authentication, you don't have any authorization yet associated to this Certificate.

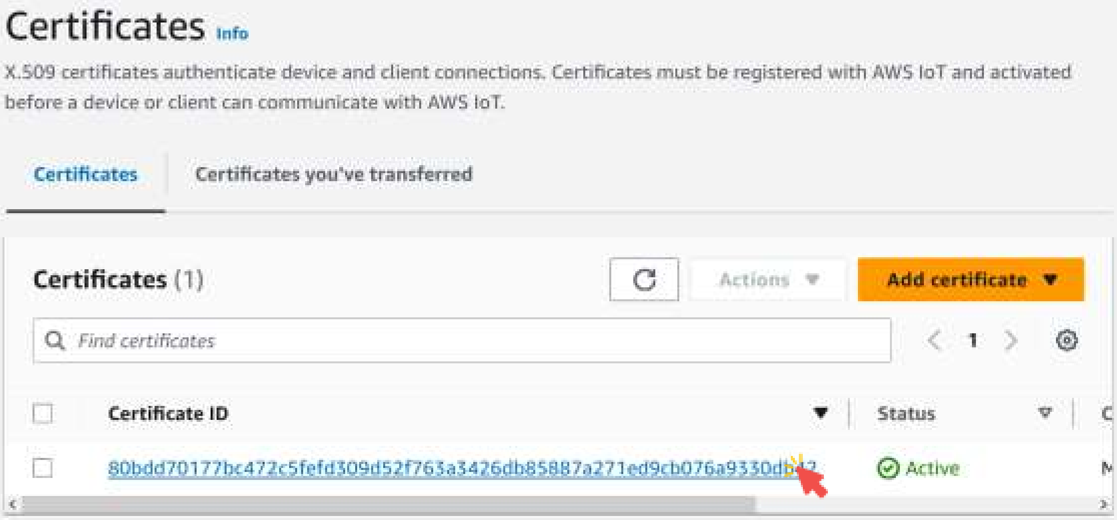


Certificates (1)

□	Certificate ID	Status
□	80bdd70177bc472c5fefd309d52f763a3426db85887a271ed9cb076a9330db42	Active

Associate Policy and Things to Your Certificate

1. Click on the certificate you just created.



Certificates

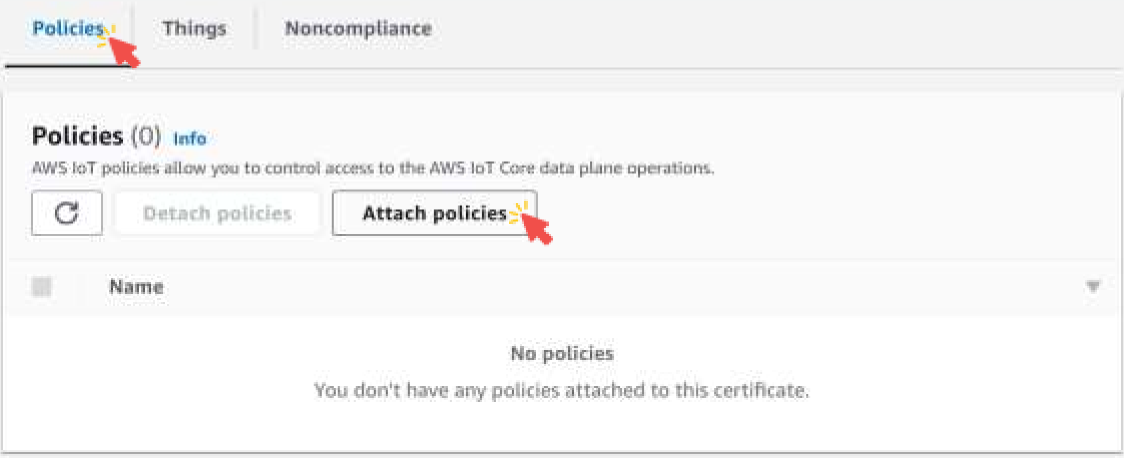
X.509 certificates authenticate device and client connections. Certificates must be registered with AWS IoT and activated before a device or client can communicate with AWS IoT.

Certificates (1)

□	Certificate ID	Status
□	80bdd70177bc472c5fefd309d52f763a3426db85887a271ed9cb076a9330db42	Active

Associate Policy and Things to Your Certificate

2. Click Policies, and then click Attach policies.



Policies (0) Info

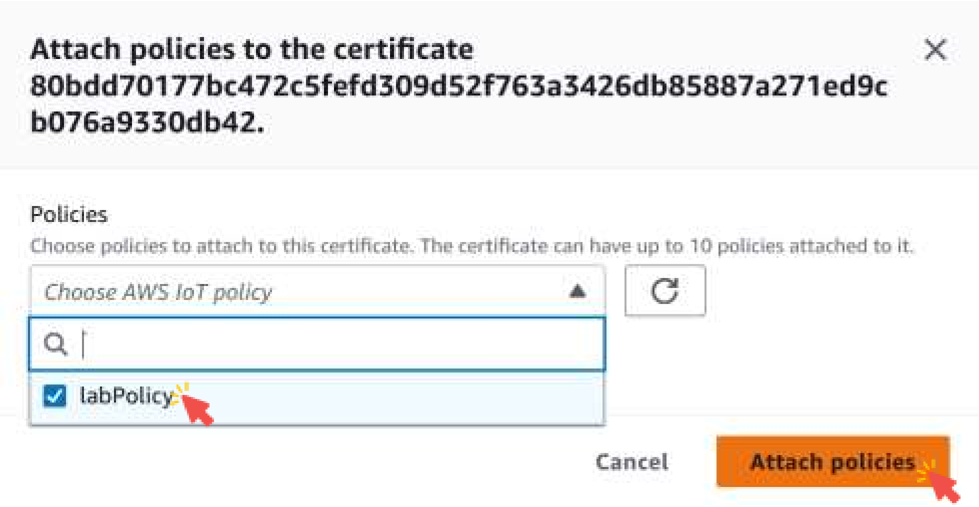
AWS IoT policies allow you to control access to the AWS IoT Core data plane operations.

No policies

You don't have any policies attached to this certificate.

Associate Policy and Things to Your Certificate

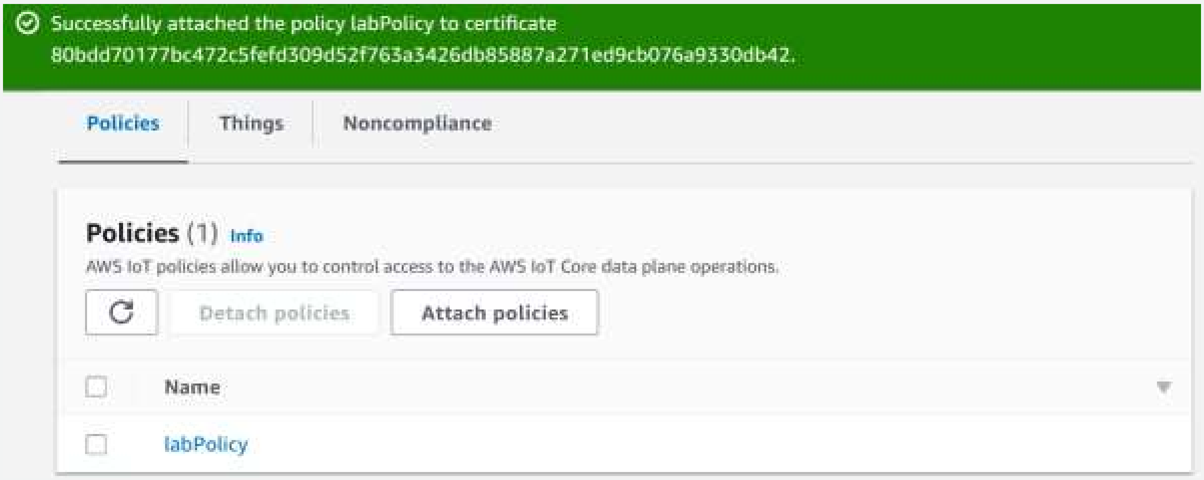
3. Put a check mark next to labPolicy and click Attach policies.


Attach policies to the certificate 80bdd70177bc472c5fefd309d52f763a3426db85887a271ed9cb076a9330db42.

Policies

Choose policies to attach to this certificate. The certificate can have up to 10 policies attached to it.

Associate Policy and Things to Your Certificate



Successfully attached the policy labPolicy to certificate 80bdd70177bc472c5fefd309d52f763a3426db85887a271ed9cb076a9330db42.

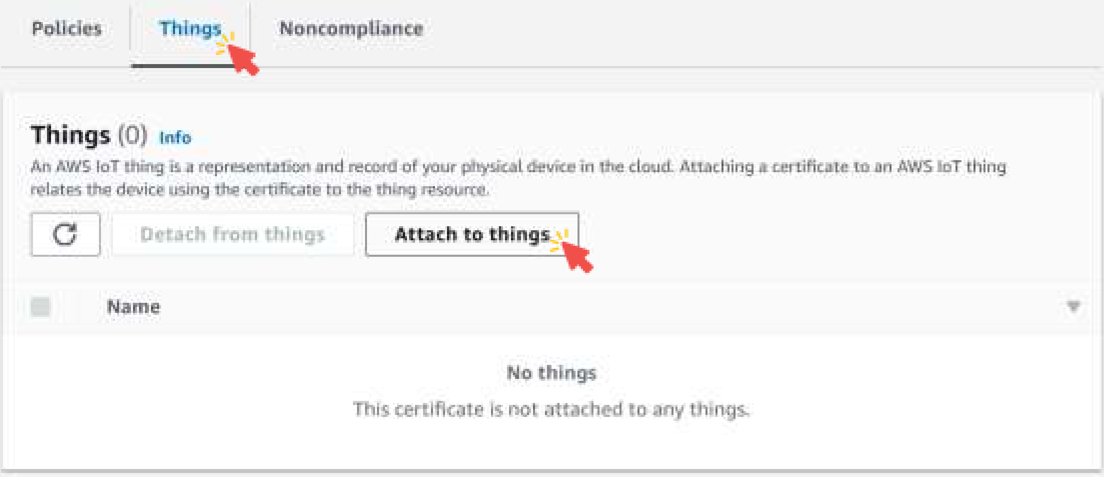
Policies (1) Info

AWS IoT policies allow you to control access to the AWS IoT Core data plane operations.

□	Name
□	labPolicy

Associate Policy and Things to Your Certificate

4. Click Things, then click Attach to things.



Things (0) Info

An AWS IoT thing is a representation and record of your physical device in the cloud. Attaching a certificate to an AWS IoT thing relates the device using the certificate to the thing resource.

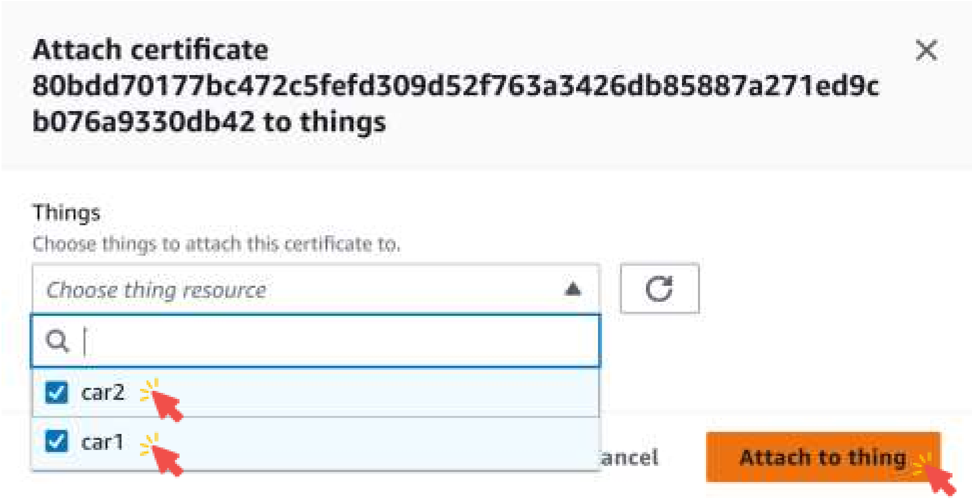
□	Name

No things

This certificate is not attached to any things.

Associate Policy and Things to Your Certificate

5. Put a check mark next to `car1` and `car2`, and click Attach to thing.



Attach certificate 80bdd70177bc472c5fefd309d52f763a3426db85887a271ed9c b076a9330db42 to things

Things

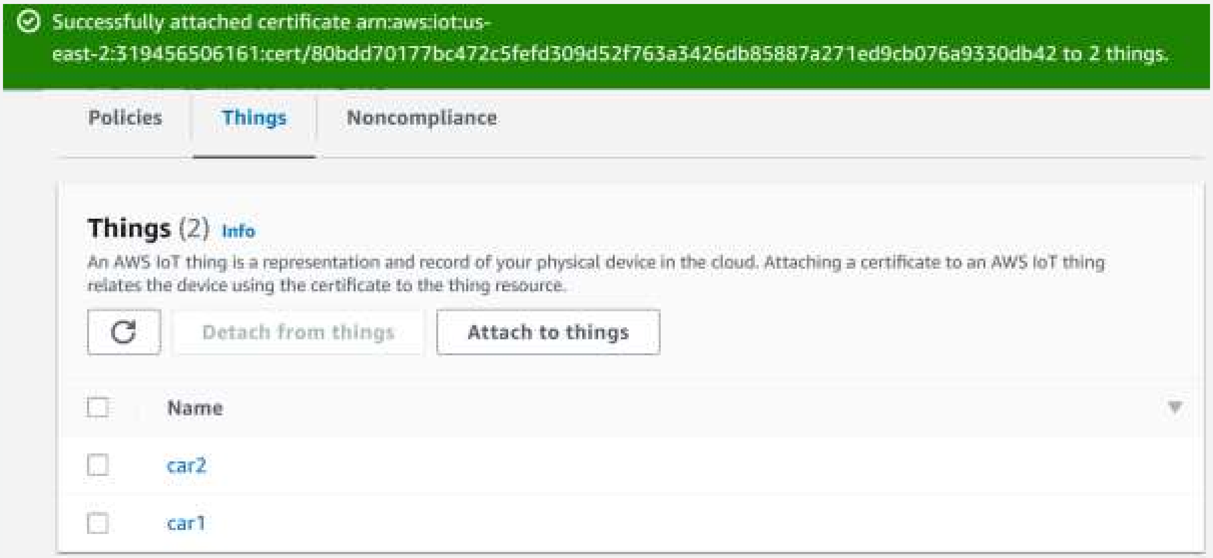
Choose things to attach this certificate to.

Choose thing resource

☑ car2

☑ car1

Associate Policy and Things to Your Certificate



Successfully attached certificate arn:aws:iot:us-east-2:319456506161:cert/80bdd70177bc472c5fefd309d52f763a3426db85887a271ed9cb076a9330db42 to 2 things.

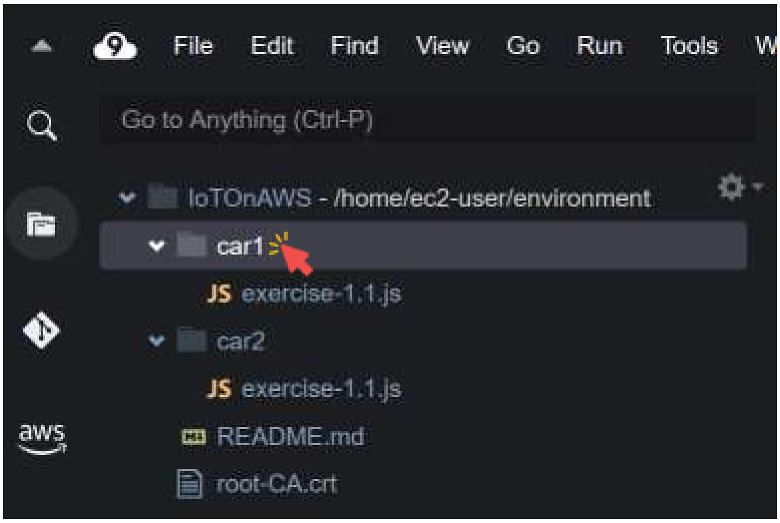
Things (2) Info

An AWS IoT thing is a representation and record of your physical device in the cloud. Attaching a certificate to an AWS IoT thing relates the device using the certificate to the thing resource.

□	Name
□	car2
□	car1

Upload Your Certificate and Private Key to Cloud9

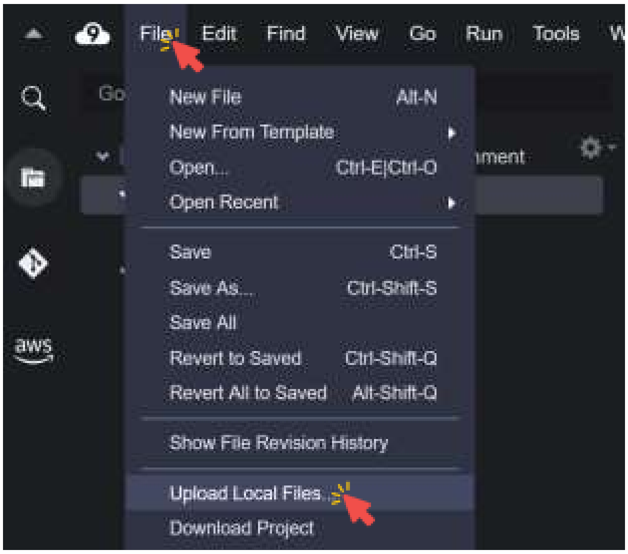
1. In the left menu, click the `car1` folder to select it.



75

Upload Your Certificate and Private Key to Cloud9

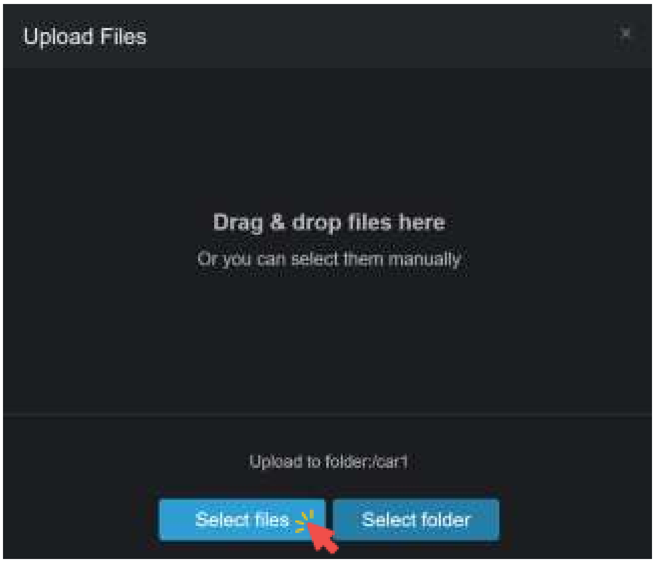
2. Click File → Upload Local Files…



76

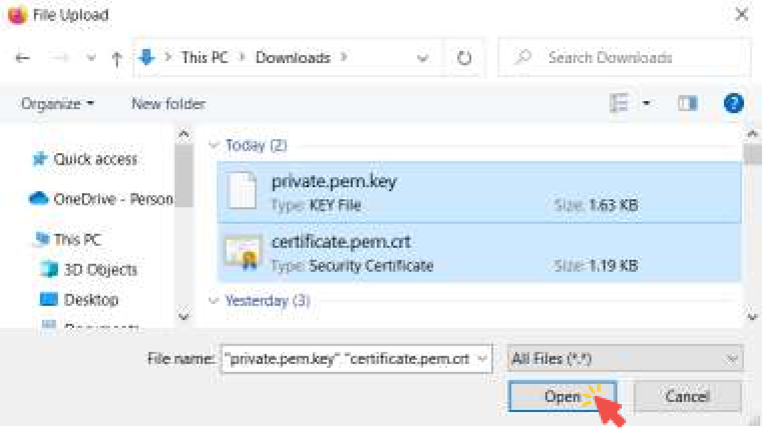
Upload Your Certificate and Private Key to Cloud9

3. Click Select Files.



Upload Your Certificate and Private Key to Cloud9

4. Choose the Certificate, and the Private key, then click Open.



private.pem.key
Type: KEY File
Size: 1.63 KB

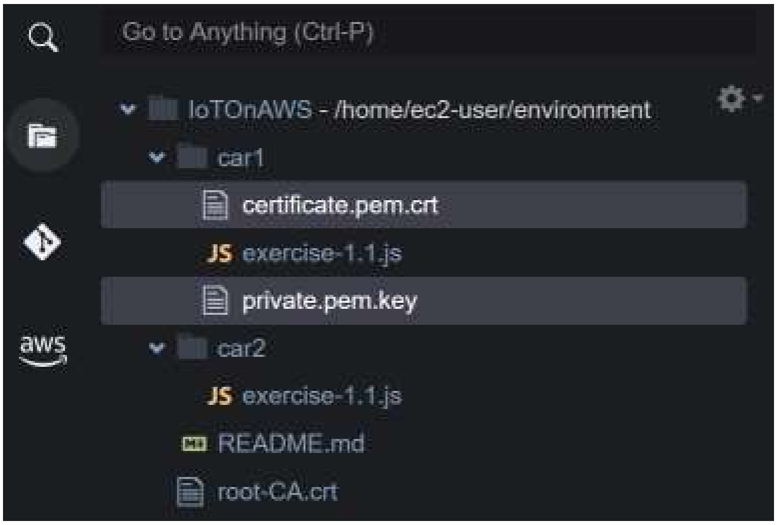
certificate.pem.crt
Type: Security Certificate
Size: 1.19 KB

File name: "private.pem.key" "certificate.pem.crt"

All Files (*.*)

Upload Your Certificate and Private Key to Cloud9

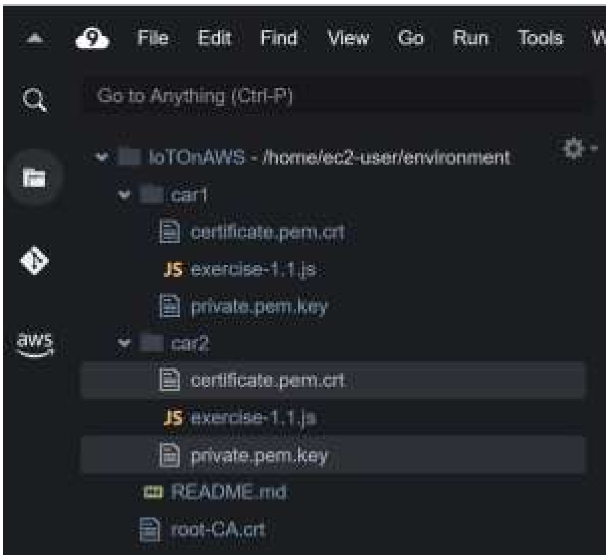
Both Certificate and Private Key should now be in the `car1` folder.



79

Upload Your Certificate and Private Key to Cloud9

5. Repeat the same steps for `car2`.

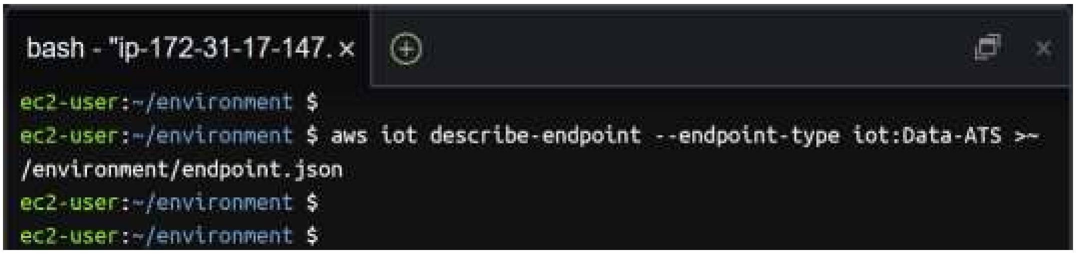


80

Execute The Code

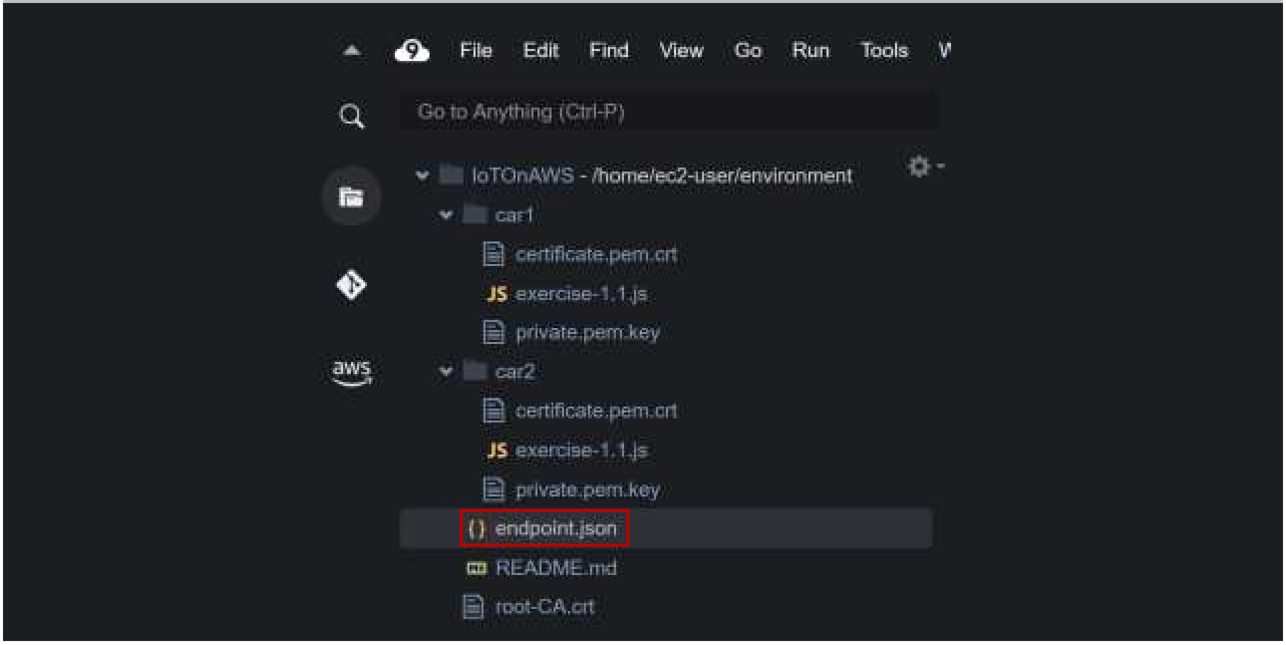
1. In the Cloud9 terminal, enter the following command to get your specific AWS IoT Endpoint which will then be saved in the endpoint.json file.

```
aws iot describe-endpoint --endpoint-type iot:Data-ATS
>~/environment/endpoint.json
```



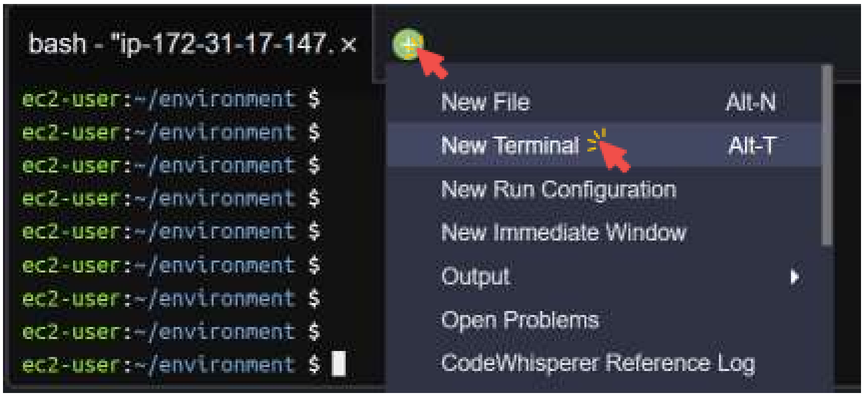
```
ec2-user:~/environment $
ec2-user:~/environment $ aws iot describe-endpoint --endpoint-type iot:Data-ATS >~
/environment/endpoint.json
ec2-user:~/environment $
ec2-user:~/environment $
```

Execute The Code



Execute The Code

2. In the Cloud9 environment, click the circled + icon that is next to your current terminal and select New Terminal.

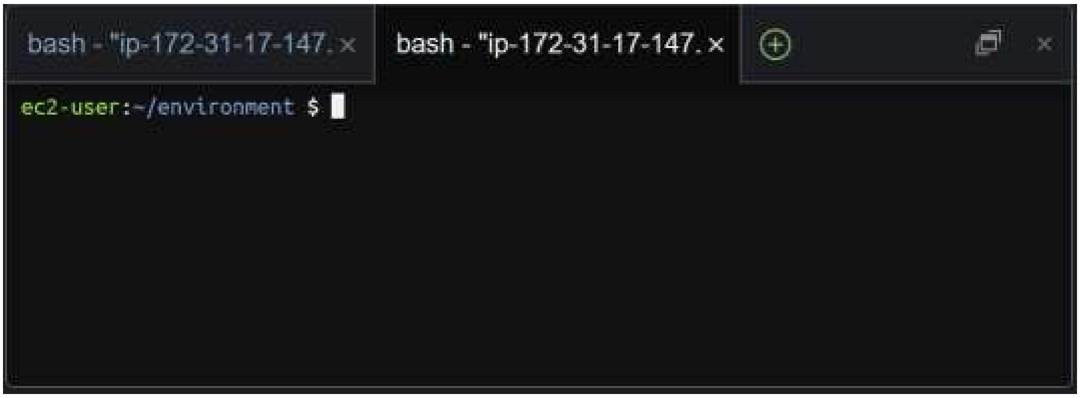


```
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
ec2-user:~/environment $
```

83

Execute The Code

- You now have 2 different terminals for the 2 cars.



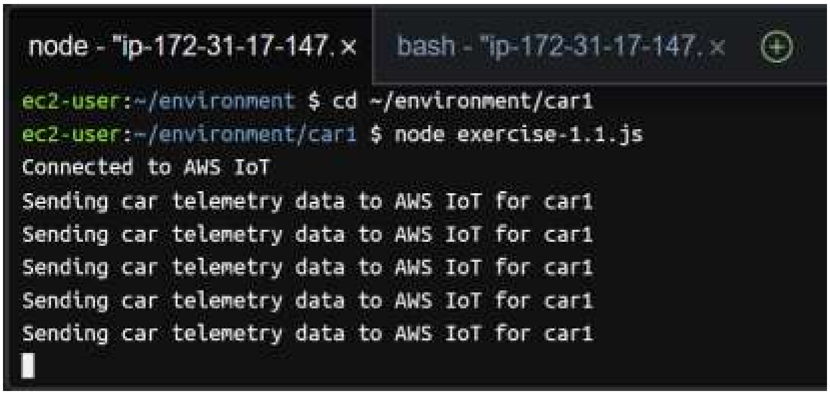
```
ec2-user:~/environment $
```

84

Execute The Code

3. In the left terminal, execute the commands to start the code for `car1`.

```
cd ~/environment/car1
node exercise-1.1.js
```



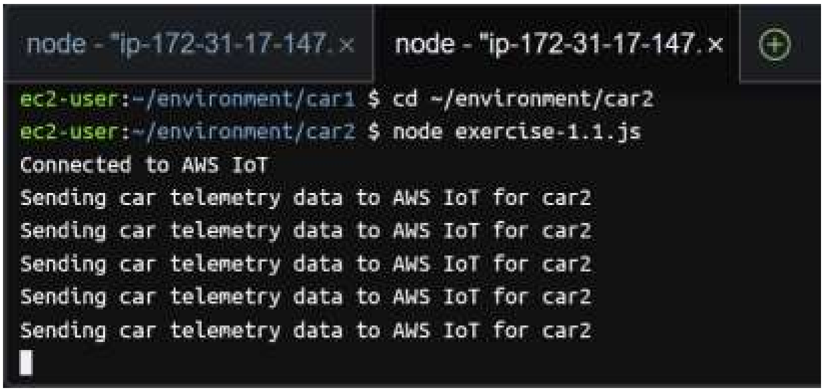
```
ec2-user:~/environment $ cd ~/environment/car1
ec2-user:~/environment/car1 $ node exercise-1.1.js
Connected to AWS IoT
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
```

85

Execute The Code

4. In the right terminal, execute the commands to start the code for `car2`.

```
cd ~/environment/car2
node exercise-1.1.js
```

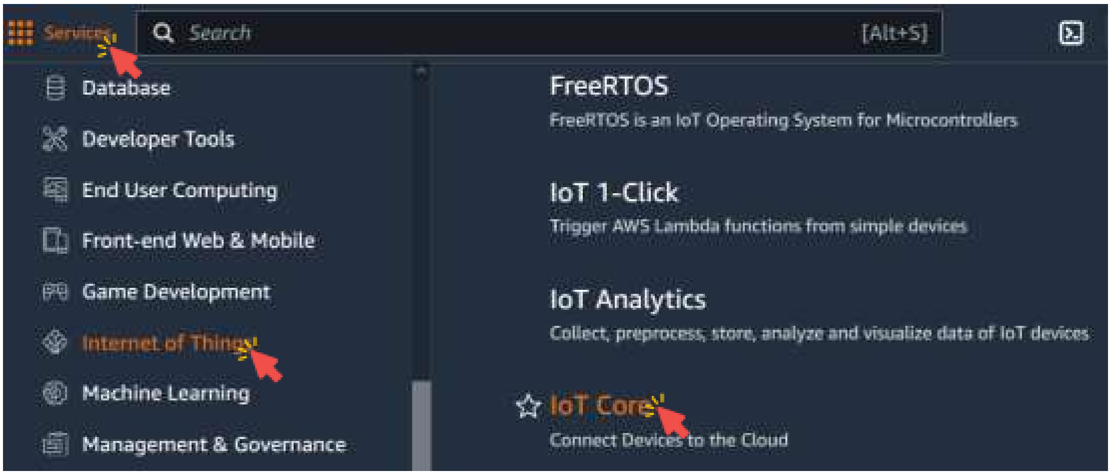


```
ec2-user:~/environment/car1 $ cd ~/environment/car2
ec2-user:~/environment/car2 $ node exercise-1.1.js
Connected to AWS IoT
Sending car telemetry data to AWS IoT for car2
Sending car telemetry data to AWS IoT for car2
Sending car telemetry data to AWS IoT for car2
Sending car telemetry data to AWS IoT for car2
Sending car telemetry data to AWS IoT for car2
```

86

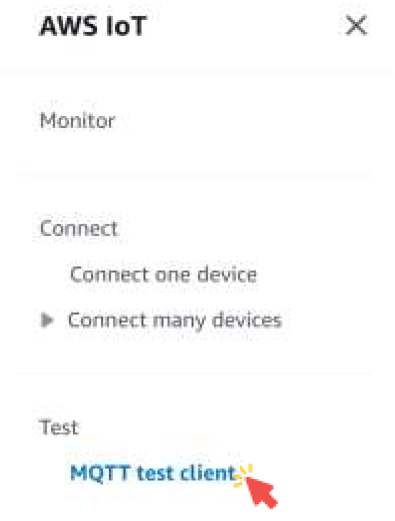
Subscribe to the `lab/telemetry` Topic

1. In the AWS Management Console, click Services, click Internet of Things and click IoT Core to open the IoT Console.


Services
Search
[Alt+S]
Database
Developer Tools
End User Computing
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
FreeRTOS
FreeRTOS is an IoT Operating System for Microcontrollers
IoT 1-Click
Trigger AWS Lambda functions from simple devices
IoT Analytics
Collect, preprocess, store, analyze and visualize data of IoT devices
IoT Core
Connect Devices to the Cloud

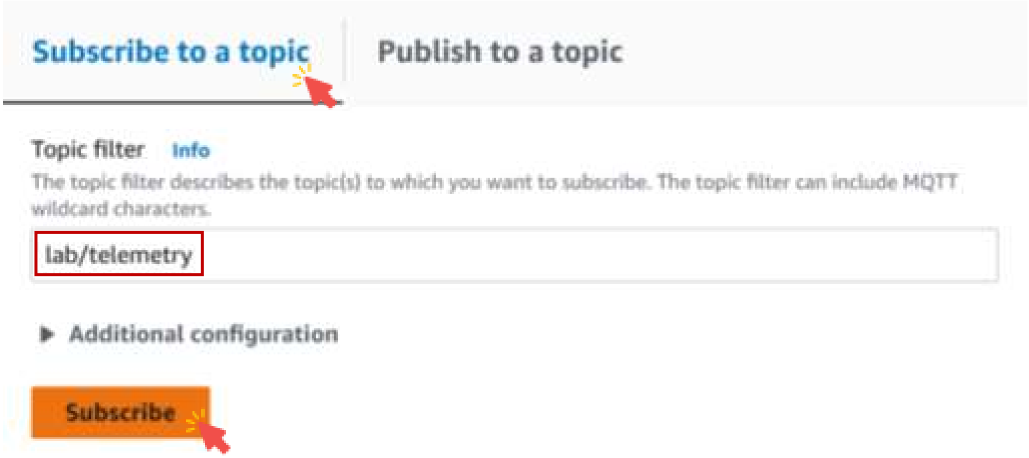
Subscribe to the `lab/telemetry` Topic

2. Click MQTT test client.



Subscribe to the `lab/telemetry` Topic

3. Click Subscribe to a topic, enter `lab/telemetry`, then click Subscribe.



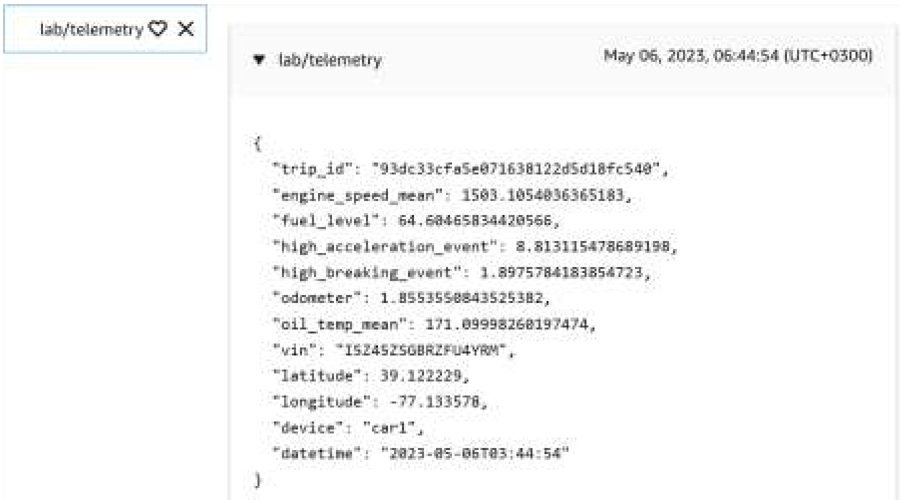
Topic filter Info

The topic filter describes the topic(s) to which you want to subscribe. The topic filter can include MQTT wildcard characters.

lab/telemetry

Subscribe to the `lab/telemetry` Topic

- You should start seeing data being published by both cars in the interface.
- You can see which car is sending data by looking at the device attribute.

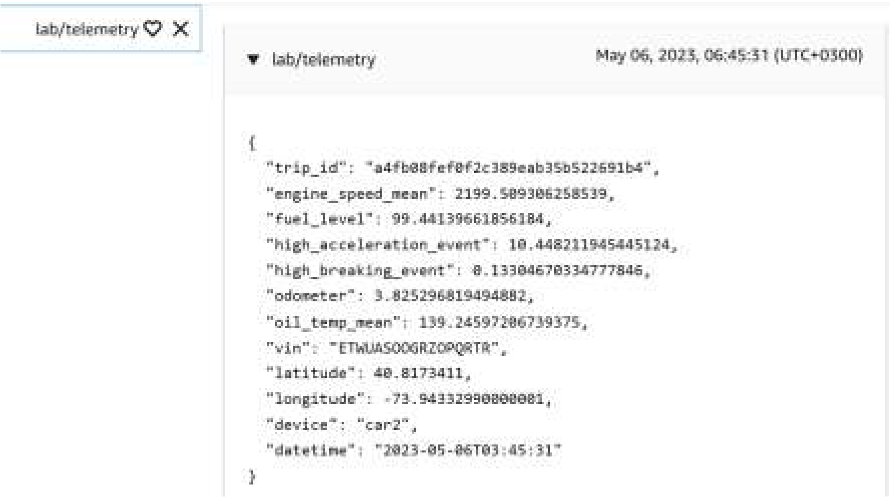


May 06, 2023, 06:44:54 (UTC+0300)

```
{
  "trip_id": "93dc33cfa5e071638122d5d18fc540",
  "engine_speed_mean": 1503.1054036365183,
  "fuel_level": 64.60465834420566,
  "high_acceleration_event": 8.813115478689198,
  "high_breaking_event": 1.8975784183854723,
  "odometer": 1.8553550843525382,
  "oil_temp_mean": 171.09998260197474,
  "vin": "1S2452SGBRZFU4YRM",
  "latitude": 39.122229,
  "longitude": -77.133578,
  "device": "car1",
  "datetime": "2023-05-06T03:44:54"
}
```

Subscribe to the lab/telemetry Topic

- You should start seeing data being published by both cars in the interface.
- You can see which car is sending data by looking at the device attribute.

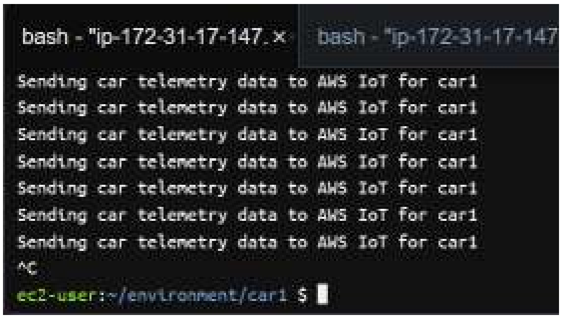


```
{
  "trip_id": "a4fb08fef0f2c389eab35b522691b4",
  "engine_speed_mean": 2199.509306258539,
  "fuel_level": 99.44139661856184,
  "high_acceleration_event": 10.448211945445124,
  "high_breaking_event": 0.13304670334777846,
  "odometer": 3.825296819494882,
  "oil_temp_mean": 139.24597206739375,
  "vin": "ETWUASOOGRZOPQRTR",
  "latitude": 40.8173411,
  "longitude": -73.94332990000081,
  "device": "car2",
  "datetime": "2023-05-06T03:45:31"
}
```

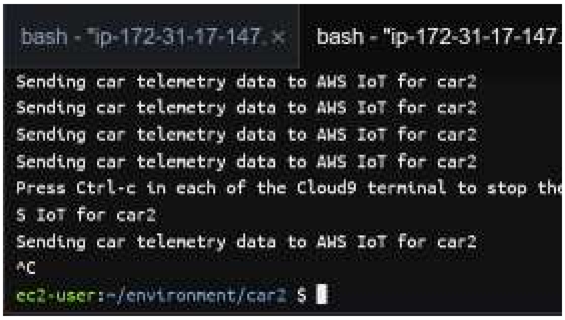
Delete The Created Resources

1. Stop the cars.

   Press `Ctrl + C` in each of the Cloud9 terminal to stop them from interacting with AWS IoT.



```
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
^C
ec2-user:~/environment/car1 $ 
```

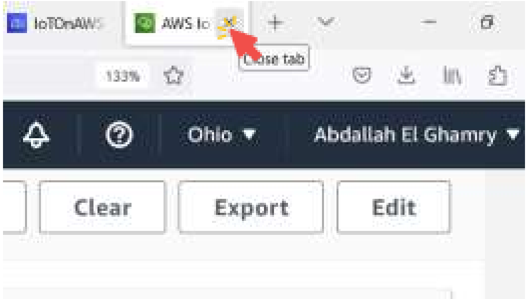


```
Sending car telemetry data to AWS IoT for car2
Sending car telemetry data to AWS IoT for car2
Sending car telemetry data to AWS IoT for car2
Sending car telemetry data to AWS IoT for car2
Press Ctrl-c in each of the Cloud9 terminal to stop the
S IoT for car2
Sending car telemetry data to AWS IoT for car2
^C
ec2-user:~/environment/car2 $ 
```

Delete The Created Resources

2. Stop the MQTT Client.

   Navigate away from the MQTT Client page to disconnect from the client.

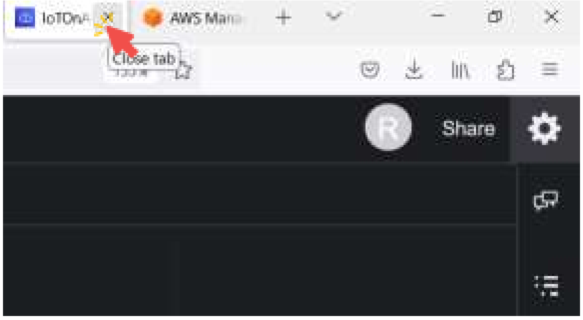


Delete The Created Resources

3. Stop the Cloud9 environment.
   - The Cloud9 will automatically shutdown after 30 minutes of inactivity.
   - For your Cloud9 environment to be considered inactive, you need to close the browser tab.
   - All the settings will be saved.

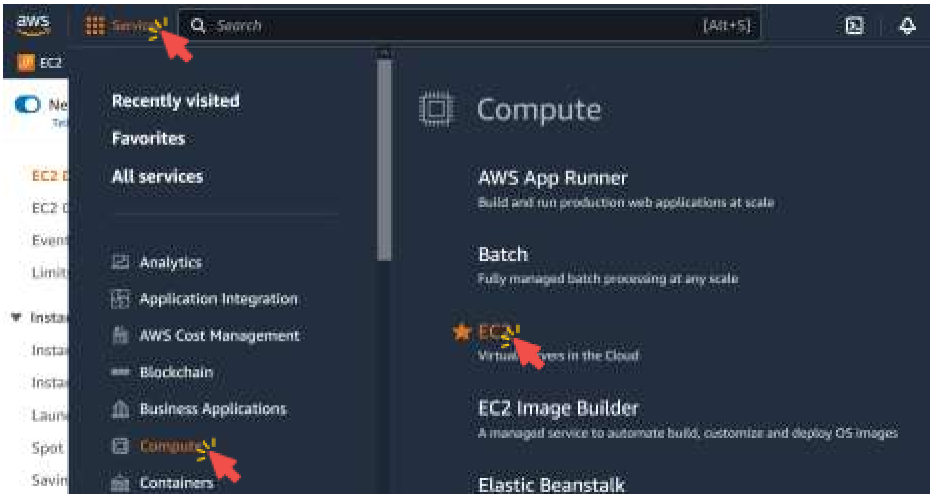
Delete The Created Resources

3.1 Close the browser tab where your environment was running.



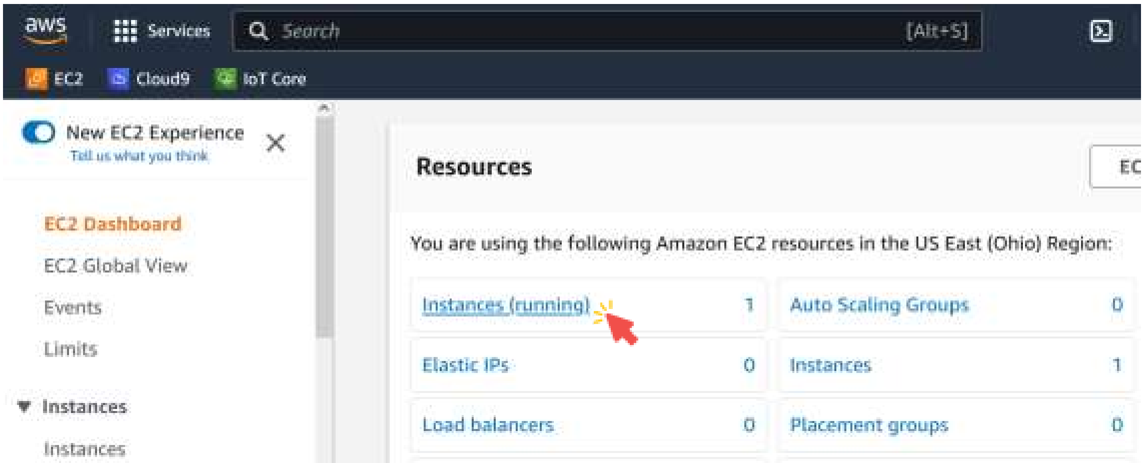
Delete The Created Resources

3.2 In the AWS Management Console, click Services, click Compute, and then click EC2 to open the EC2 console.



Delete The Created Resources

3.3 Click Instances in the left menu.

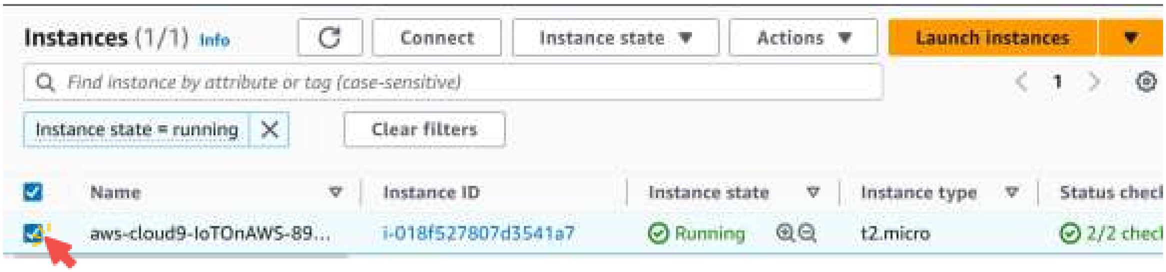


You are using the following Amazon EC2 resources in the US East (Ohio) Region:

Instances (running)	1	Auto Scaling Groups	0
Elastic IPs	0	Instances	1
Load balancers	0	Placement groups	0

Delete The Created Resources

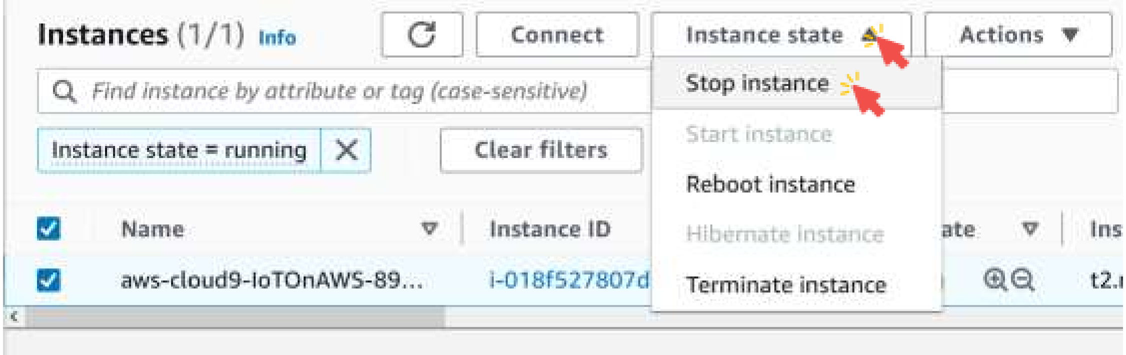
3.4 Select the EC2 Instance that starts with `aws-cloud9-IoTOnAWS`.



☑	Name	Instance ID	Instance state	Instance type	Status checl
☑	aws-cloud9-IoTOnAWS-89...	i-018f527807d3541a7	Running	t2.micro	2/2 checl

Delete The Created Resources

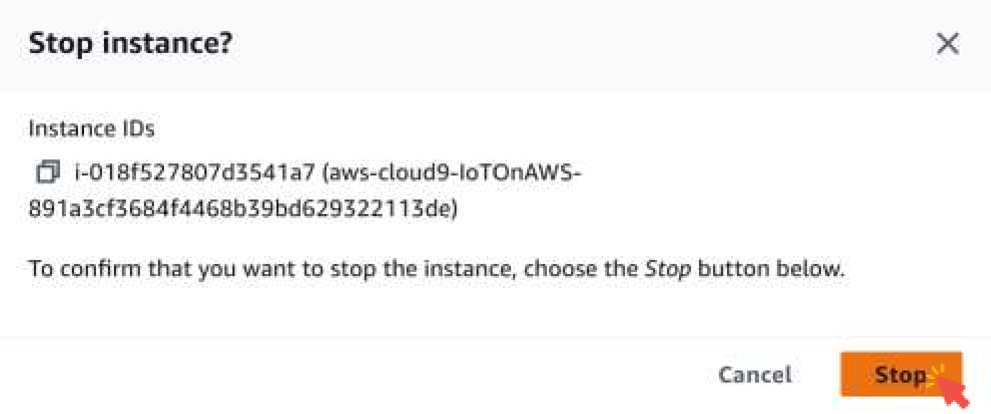
3.5 Click Instance State → Stop instance.



99

Delete The Created Resources

3.6 Click Stop.



Stop instance?

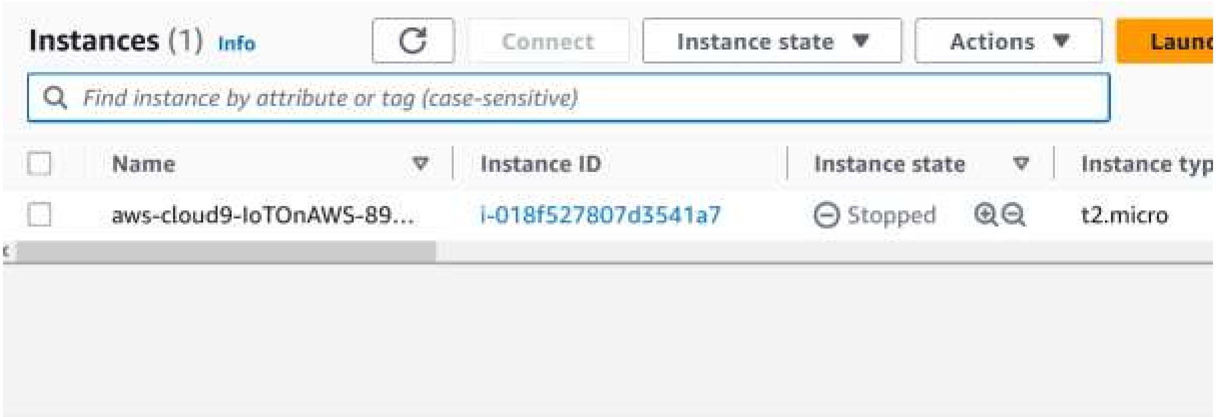
Instance IDs

i-018f527807d3541a7 (aws-cloud9-IoTOnAWS-891a3cf3684f4468b39bd629322113de)

To confirm that you want to stop the instance, choose the *Stop* button below.

100

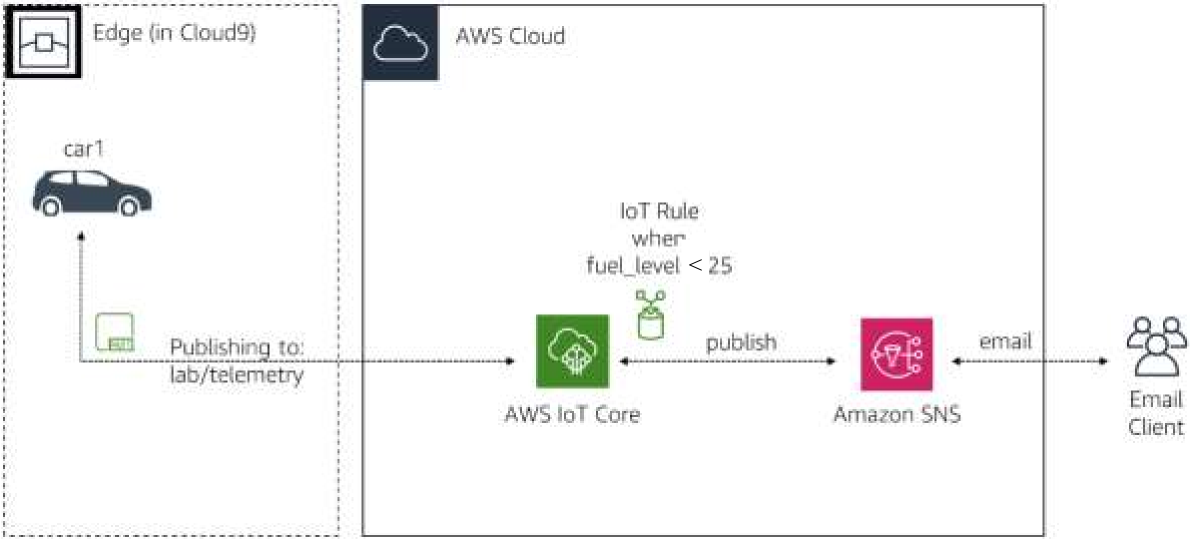
Delete The Created Resources



□	Name	Instance ID	Instance state	Instance typ
□	aws-cloud9-IoTOnAWS-89...	i-018f527807d3541a7	⊖ Stopped	t2.micro

101

AWS IoT Rules



102

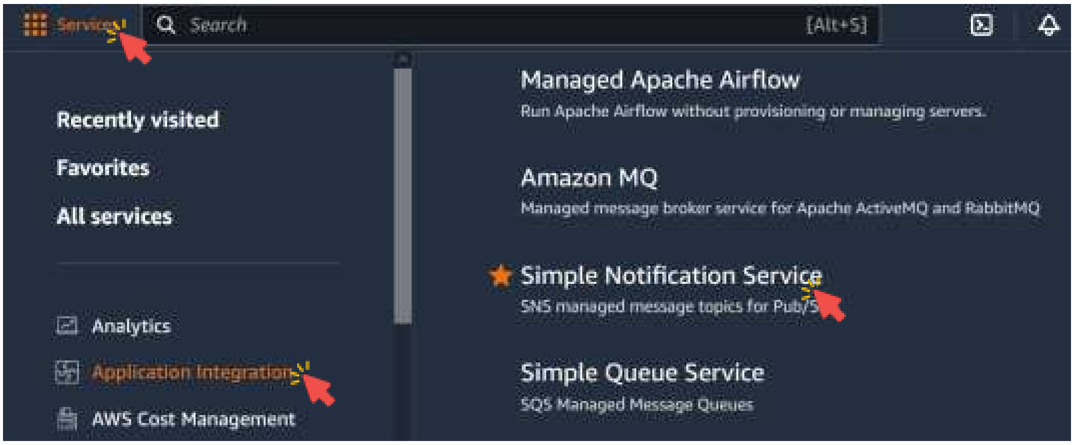
AWS IoT Rules

- In this exercise, you will create a trigger that sends you an email when the fuel level attribute of the car is lower than 25%.
- To do this, you will create a Simple Notification Service (SNS) Topic and subscribe to it via your email.
- You will need to give permission to the AWS IoT service to publish a notification to SNS.
- Finally, you will create an IoT Rule looking for the fuel level using a SQL query publishing to your SNS Topic when it matches.

103

Create a Simple Notification Service Topic

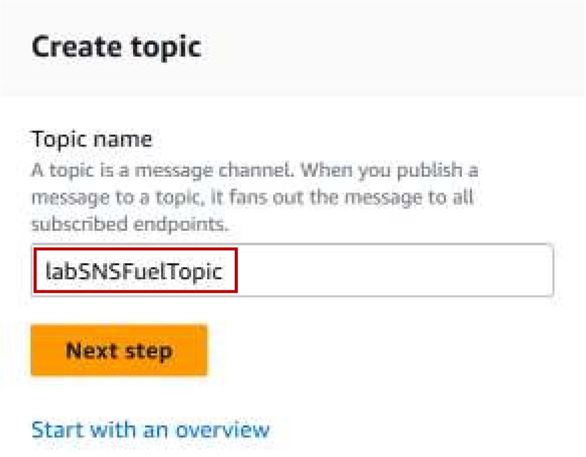
1. In the AWS Management Console, click Services, click Application Integration, and then click Simple Notification Service to go to the SNS.


Services
Search
[Alt+S]
Recently visited
Favorites
All services
Analytics
Application Integration
AWS Cost Management
Managed Apache Airflow
Run Apache Airflow without provisioning or managing servers.
Amazon MQ
Managed message broker service for Apache ActiveMQ and RabbitMQ
Simple Notification Service
SNS managed message topics for Pub/S
Simple Queue Service
SQS Managed Message Queues

104

Create a Simple Notification Service Topic

2. For the Topic name enter `labSNSFuelTopic`.



Create topic

Topic name

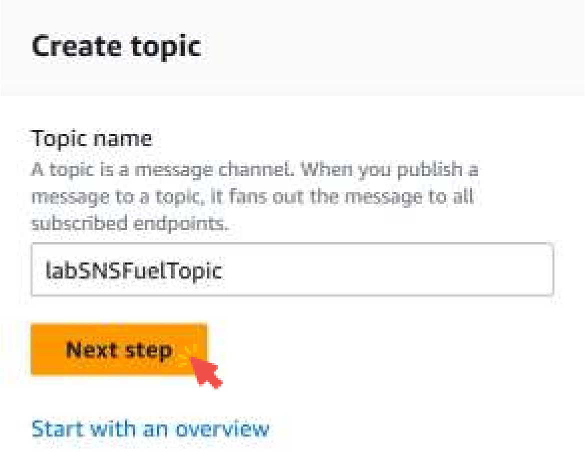
A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.

labSNSFuelTopic

105

Create a Simple Notification Service Topic

3. Click Next step.



Create topic

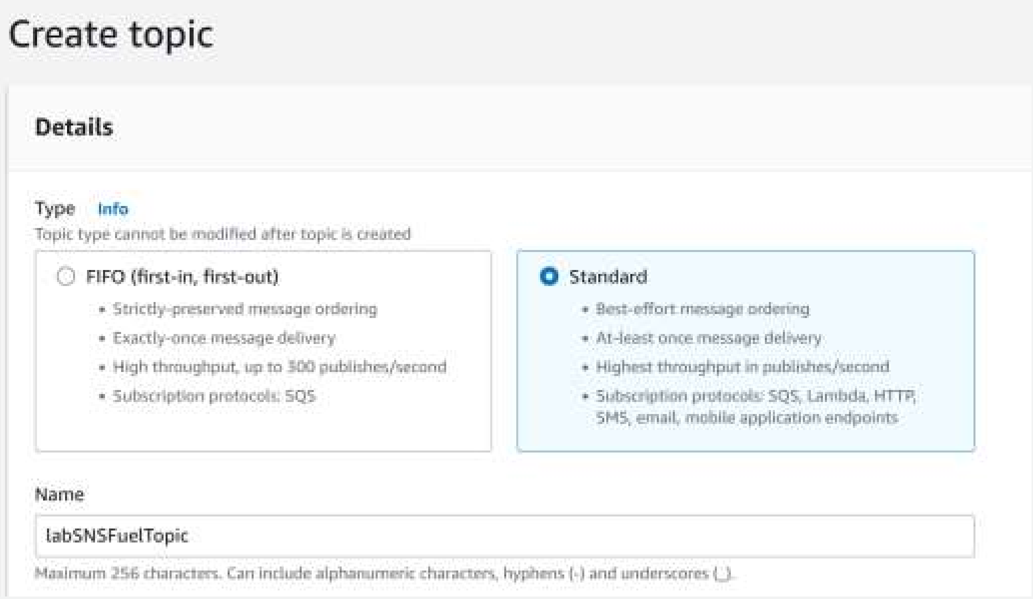
Topic name

A topic is a message channel. When you publish a message to a topic, it fans out the message to all subscribed endpoints.

labSNSFuelTopic

106

Create a Simple Notification Service Topic



Create topic

Details

Type Info

Topic type cannot be modified after topic is created

○ FIFO (first-in, first-out)

- Strictly-preserved message ordering
- Exactly-once message delivery
- High throughput, up to 300 publishes/second
- Subscription protocols: SQS

● Standard

- Best-effort message ordering
- At-least once message delivery
- Highest throughput in publishes/second
- Subscription protocols: SQS, Lambda, HTTP, SMS, email, mobile application endpoints

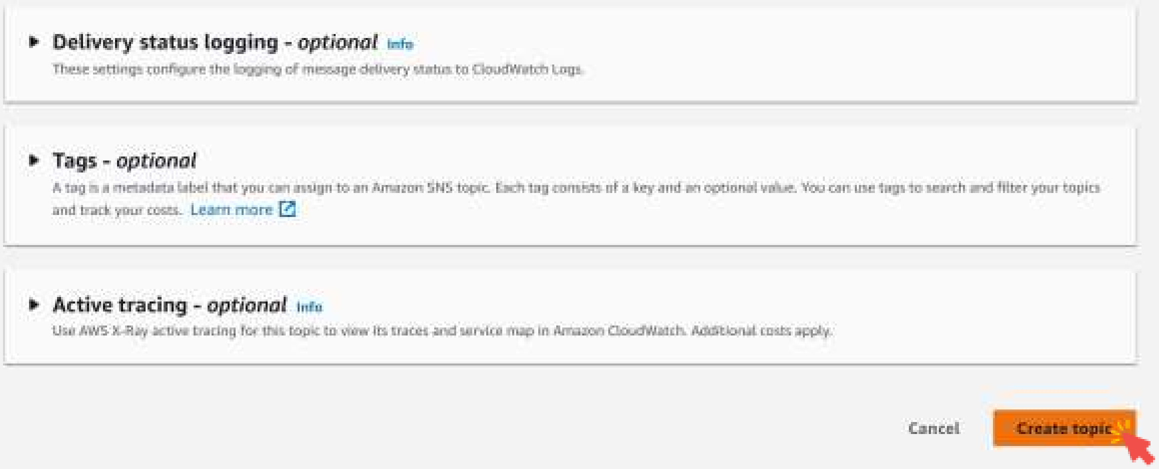
Name

labSNSFuelTopic

Maximum 256 characters. Can include alphanumeric characters, hyphens (-) and underscores (_).

Create a Simple Notification Service Topic

4. Click Create topic.



▸ **Delivery status logging** - *optional* Info

These settings configure the logging of message delivery status to CloudWatch Logs.

▸ **Tags** - *optional*

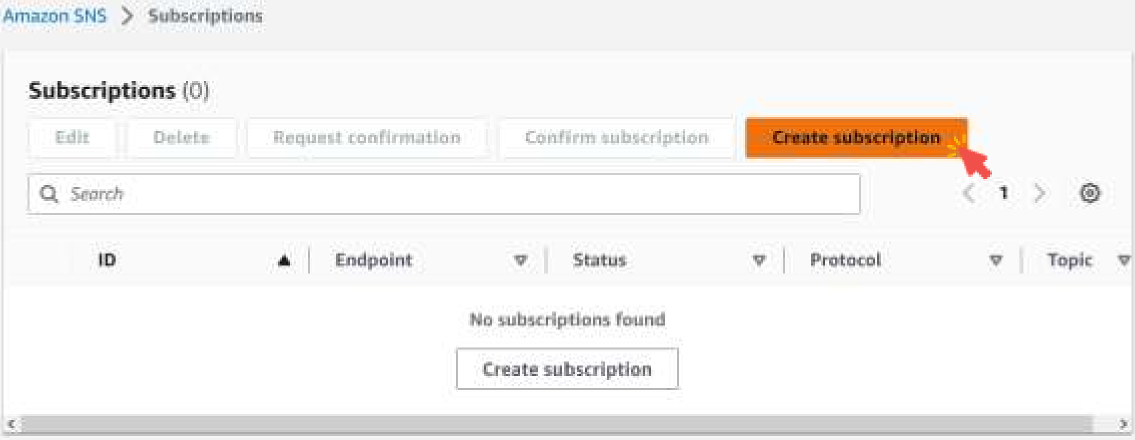
A tag is a metadata label that you can assign to an Amazon SNS topic. Each tag consists of a key and an optional value. You can use tags to search and filter your topics and track your costs. Learn more

▸ **Active tracing** - *optional* Info

Use AWS X-Ray active tracing for this topic to view its traces and service map in Amazon CloudWatch. Additional costs apply.

Create a Simple Notification Service Topic

5. Click Create subscription.



Amazon SNS > Subscriptions

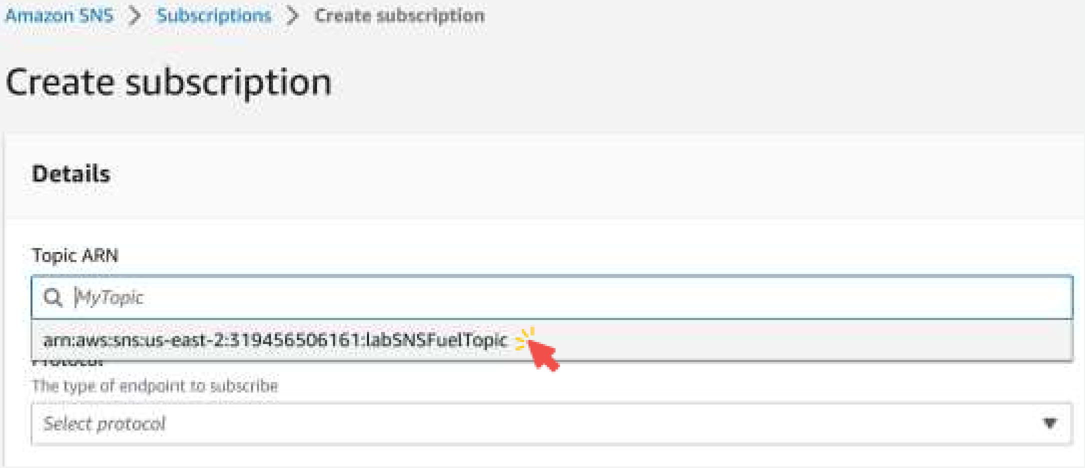
Subscriptions (0)

No subscriptions found

109

Create a Simple Notification Service Topic

6. Select the topic `labSNSFuelTopic`.



Amazon SNS > Subscriptions > Create subscription

Create subscription

Details

Topic ARN: MyTopic

arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic

Protocol

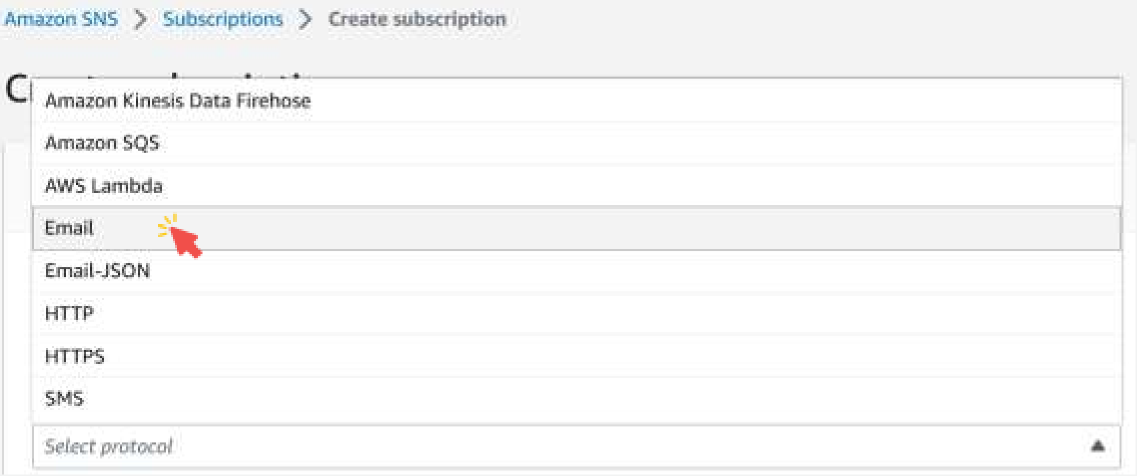
The type of endpoint to subscribe

Select protocol

110

Create a Simple Notification Service Topic

7. Select the protocol Email.



Amazon SNS > Subscriptions > Create subscription

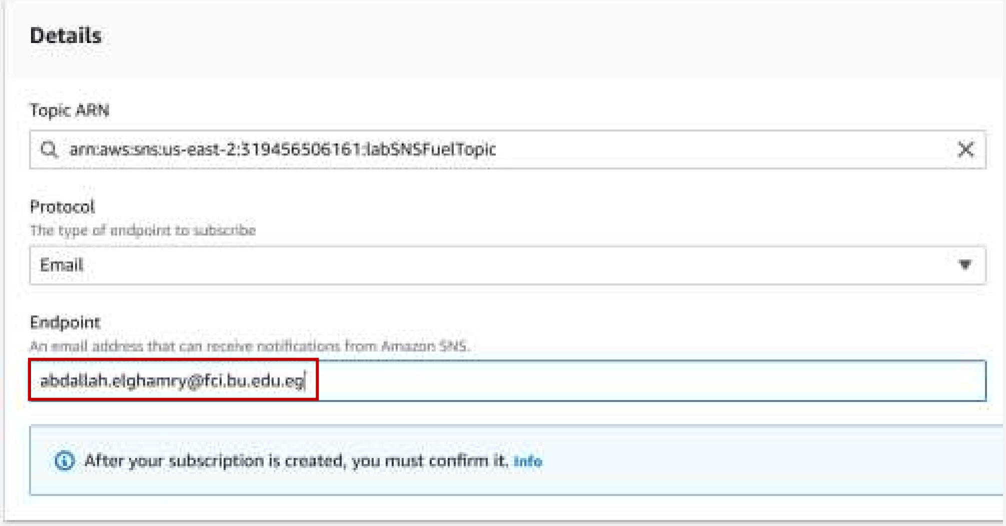
- Amazon Kinesis Data Firehose
- Amazon SQS
- AWS Lambda
- Email
- Email-JSON
- HTTP
- HTTPS
- SMS

Select protocol ▲

111

Create a Simple Notification Service Topic

8. Write the subscribed email in the endpoint.



Details

Topic ARN

arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic

Protocol

The type of endpoint to subscribe

Email ▼

Endpoint

An email address that can receive notifications from Amazon SNS.

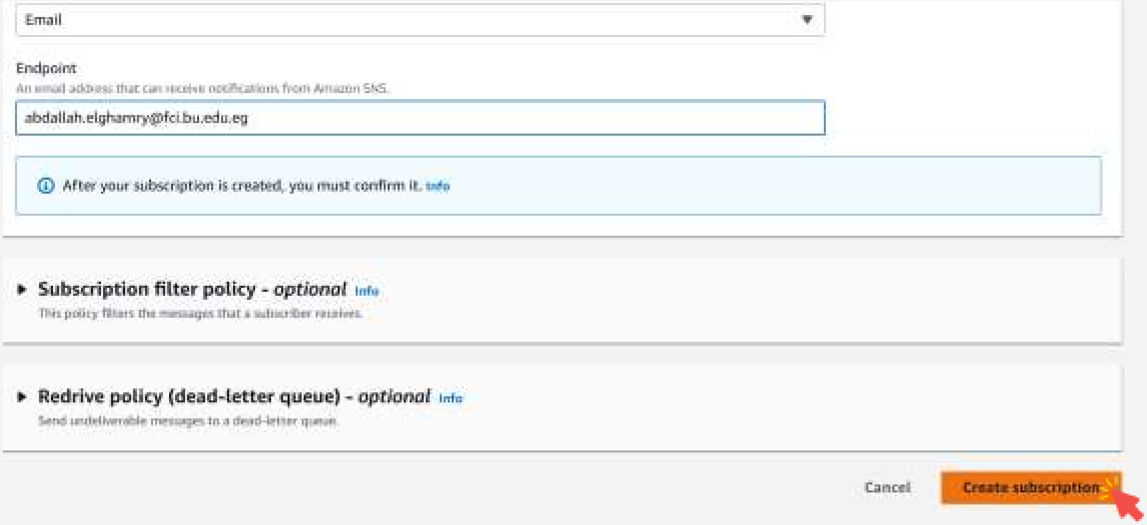
abdallah.elghamry@fci.bu.edu.eg

After your subscription is created, you must confirm it. Info

112

Create a Simple Notification Service Topic

9. Click Create subscription.



Email

Endpoint
An email address that can receive notifications from Amazon SNS.
abdallah.elghamry@fci.bu.edu.eg

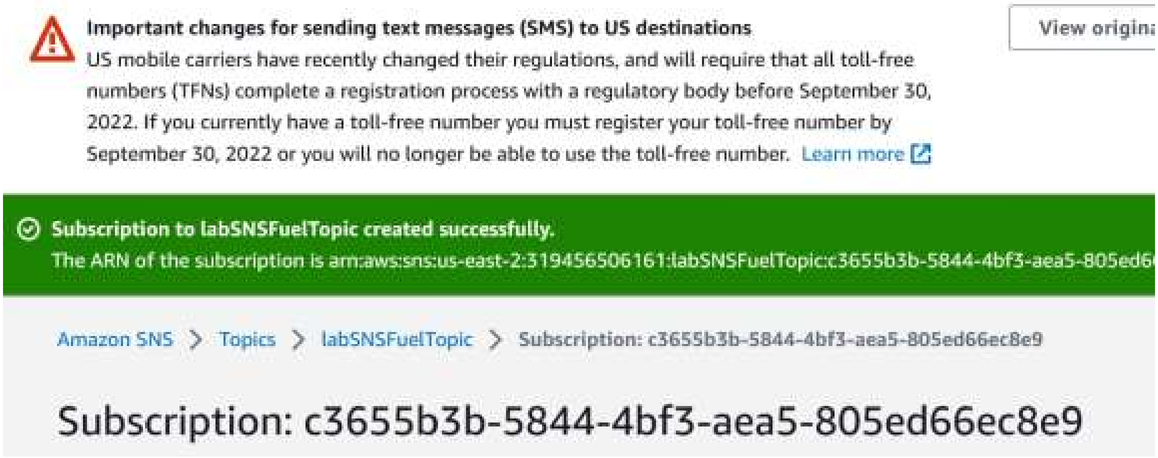
ⓘ After your subscription is created, you must confirm it. Info

▸ Subscription filter policy - *optional* Info
This policy filters the messages that a subscriber receives.

▸ Redrive policy (dead-letter queue) - *optional* Info
Send undeliverable messages to a dead-letter queue.

Create a Simple Notification Service Topic

- Within a few minutes, you will receive an email to the address you have specified



Important changes for sending text messages (SMS) to US destinations
US mobile carriers have recently changed their regulations, and will require that all toll-free numbers (TFNs) complete a registration process with a regulatory body before September 30, 2022. If you currently have a toll-free number you must register your toll-free number by September 30, 2022 or you will no longer be able to use the toll-free number. Learn more

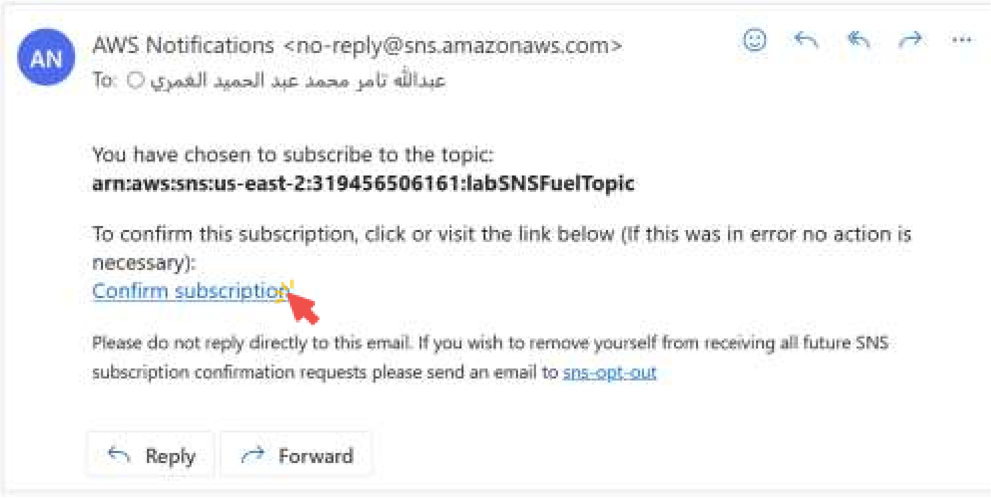
Subscription to labSNSFuelTopic created successfully.
The ARN of the subscription is arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic:c3655b3b-5844-4bf3-aea5-805ed6

Amazon SNS > Topics > labSNSFuelTopic > Subscription: c3655b3b-5844-4bf3-aea5-805ed66ec8e9

Subscription: c3655b3b-5844-4bf3-aea5-805ed66ec8e9

Create a Simple Notification Service Topic

10. Open that email and click the Confirm subscription.



AWS Notifications <no-reply@sns.amazonaws.com>

To: ○ عبدالله تامر محمد عبد الحميد الغمري

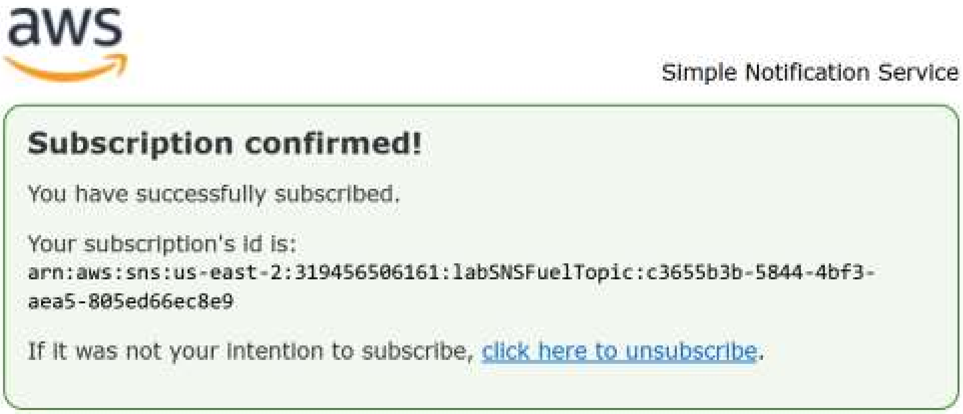
You have chosen to subscribe to the topic:
arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic

To confirm this subscription, click or visit the link below (If this was in error no action is necessary):
Confirm subscription

Please do not reply directly to this email. If you wish to remove yourself from receiving all future SNS subscription confirmation requests please send an email to sns-opt-out

Create a Simple Notification Service Topic

- You have now successfully created an SNS Topic and subscribed your email address to it.



Subscription confirmed!

You have successfully subscribed.

Your subscription's id is:
arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic:c3655b3b-5844-4bf3-aea5-805ed66ec8e9

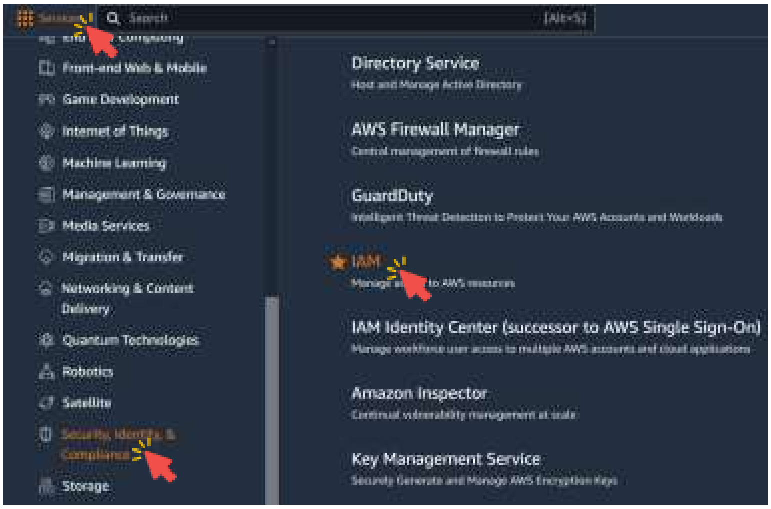
If it was not your intention to subscribe, click here to unsubscribe.

Create an IAM Role

- For the AWS IoT service to be able to publish a new message on the SNS Topic you created in the previous section, an IAM Role must be created.
- IAM stands for Identity and Access Management.

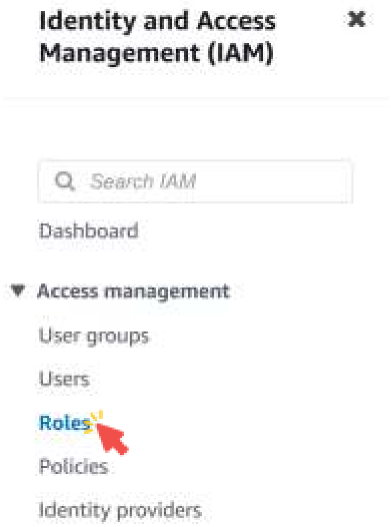
Create an IAM Role

1. In the AWS Management Console, click Services, click Security, Identity, & Compliance, and then click IAM to go to the IAM dashboard.


Services
Search
[Alt+S]
End User Computing
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
Media Services
Migration & Transfer
Networking & Content Delivery
Quantum Technologies
Robotics
Satellite
Security, Identity, & Compliance
Storage
Directory Service
Host and Manage Active Directory
AWS Firewall Manager
Central management of firewall rules
GuardDuty
Intelligent Threat Detection to Protect Your AWS Accounts and Workloads
IAM
Manage a to AWS resources
IAM Identity Center (successor to AWS Single Sign-On)
Manage workforce user access to multiple AWS accounts and cloud applications
Amazon Inspector
Continual vulnerability management at scale
Key Management Service
Securely Generate and Manage AWS Encryption Keys

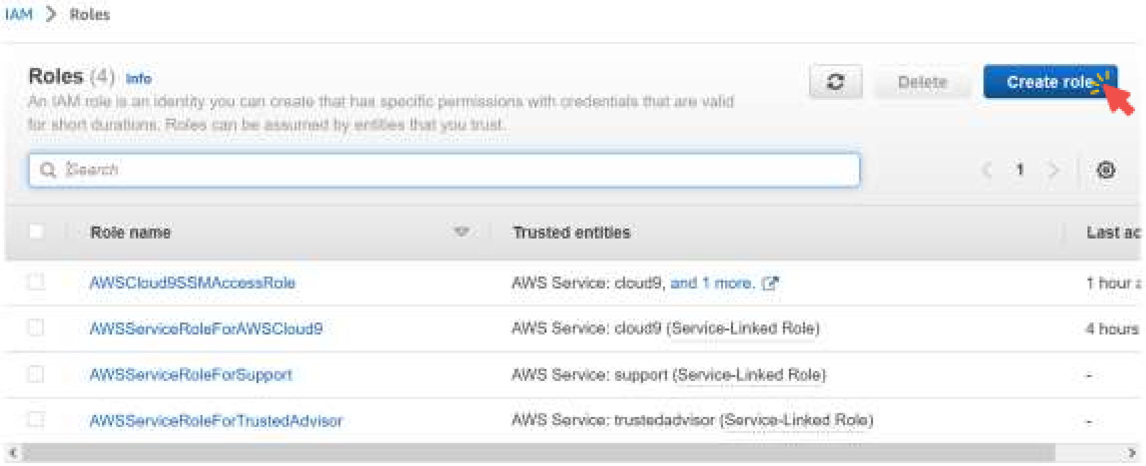
Create an IAM Role

2. In the left navigation menu, click Roles.



Create an IAM Role

3. Click Create role.



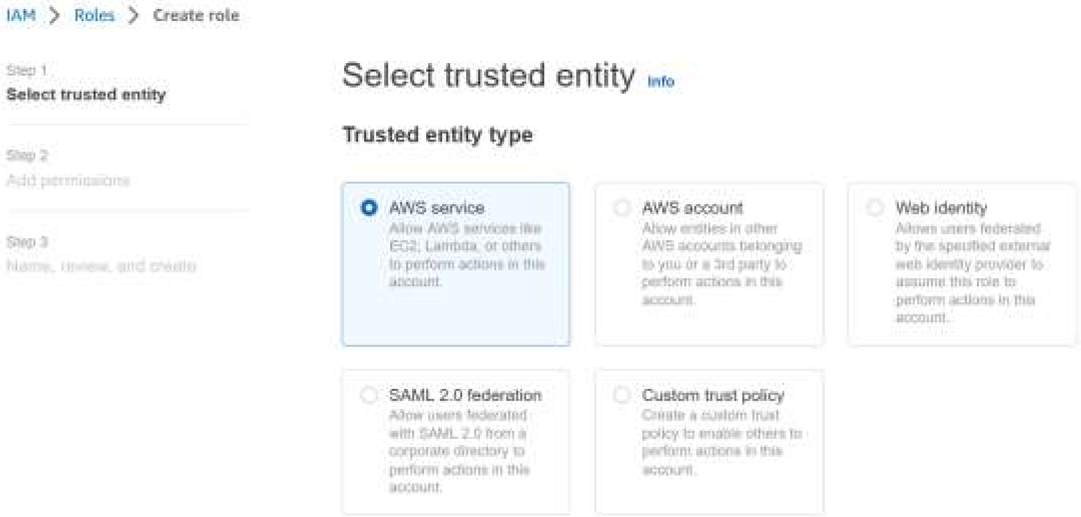
Roles (4) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

□	Role name	Trusted entities	Last ac
□	AWSCloud9SSMAccessRole	AWS Service: cloud9, and 1 more.	1 hour a
□	AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked Role)	4 hours
□	AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)	-
□	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked Role)	-

Create an IAM Role

4. Select AWS service as a trusted entity.



IAM > Roles > Create role

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Select trusted entity Info

Trusted entity type

● AWS service
Allow AWS services like EC2, Lambda, or others to perform actions in this account.

○ AWS account
Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

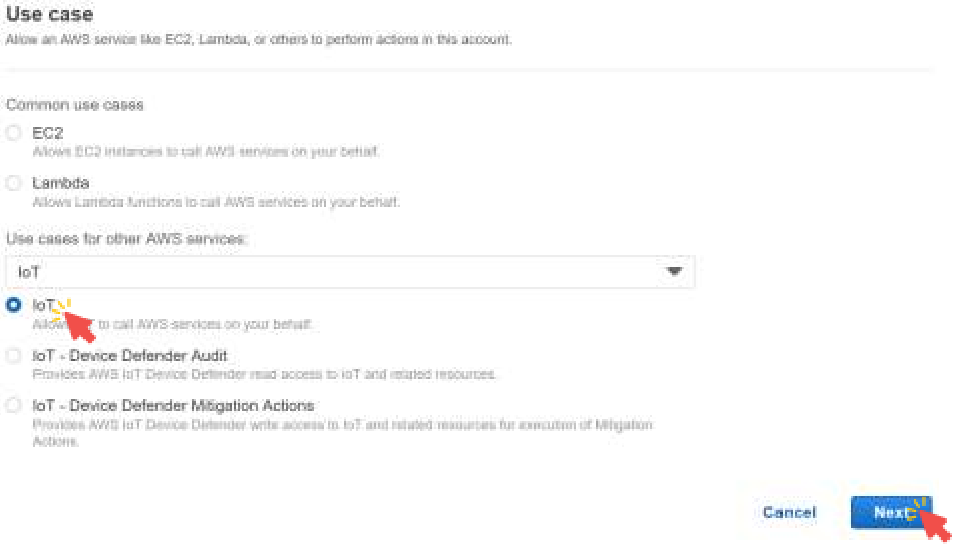
○ Web identity
Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

○ SAML 2.0 federation
Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

○ Custom trust policy
Create a custom trust policy to enable others to perform actions in this account.

Create an IAM Role

5. Select IoT from Use cases for other AWS services, and click Next.



Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Common use cases

○ EC2
Allows EC2 instances to call AWS services on your behalf.

○ Lambda
Allows Lambda functions to call AWS services on your behalf.

Use cases for other AWS services:

IoT

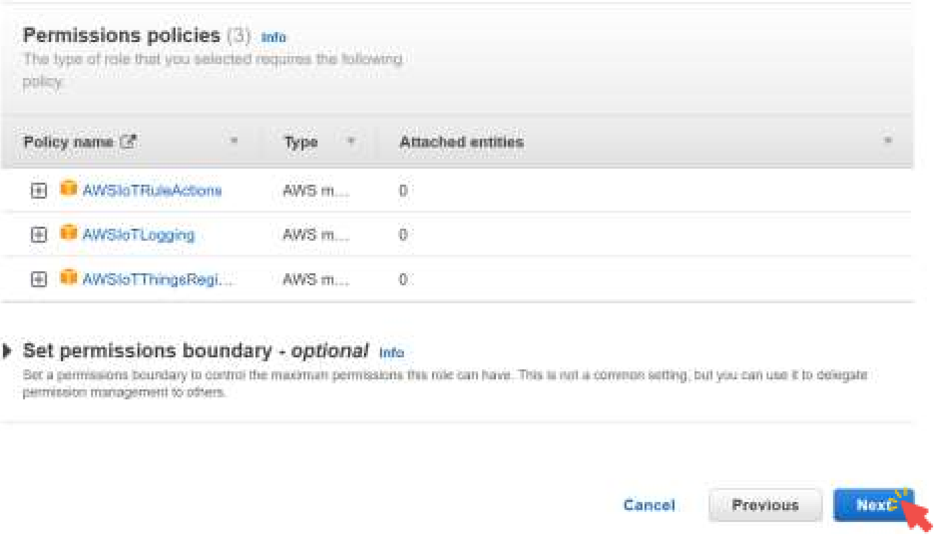
● IoT
Allow to call AWS services on your behalf.

○ IoT - Device Defender Audit
Provides AWS IoT Device Defender read access to IoT and related resources.

○ IoT - Device Defender Mitigation Actions
Provides AWS IoT Device Defender write access to IoT and related resources for execution of Mitigation Actions.

Create an IAM Role

6. Click Next in Permissions policies.



Permissions policies (3) Info

The type of role that you selected requires the following policy.

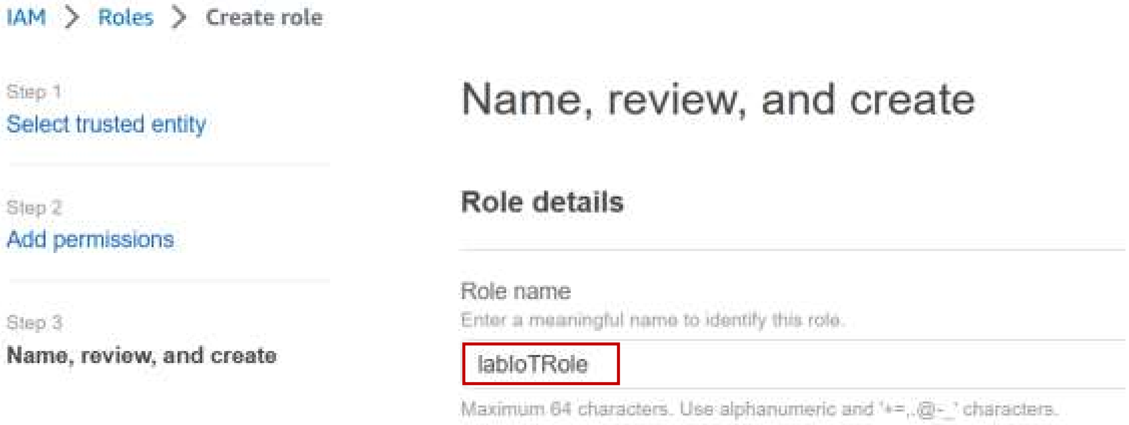
Policy name	Type	Attached entities
⊞ AWSIoTRuleActions	AWS m...	0
⊞ AWSIoTLogging	AWS m...	0
⊞ AWSIoTThingsRegi...	AWS m...	0

▸ **Set permissions boundary - *optional*** Info

Set a permissions boundary to control the maximum permissions this role can have. This is not a common setting, but you can use it to delegate permission management to others.

Create an IAM Role

7. For Role name, enter `labIoTRole`.



IAM > Roles > Create role

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Name, review, and create

Role details

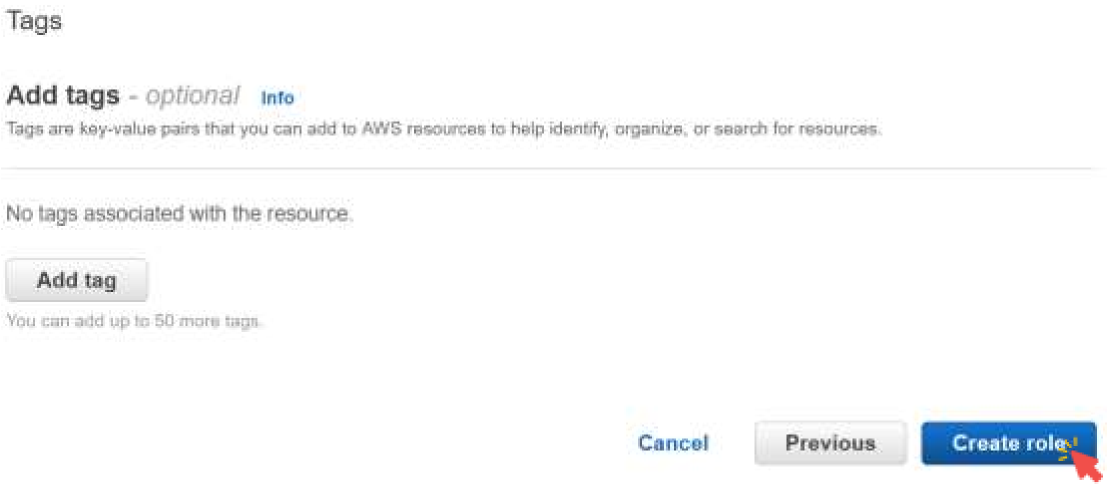
Role name
Enter a meaningful name to identify this role.

labIoTRole

Maximum 64 characters. Use alphanumeric and '+=,.@-_' characters.

Create an IAM Role

8. Click Create role.



Tags

Add tags - *optional* Info

Tags are key-value pairs that you can add to AWS resources to help identify, organize, or search for resources.

No tags associated with the resource.

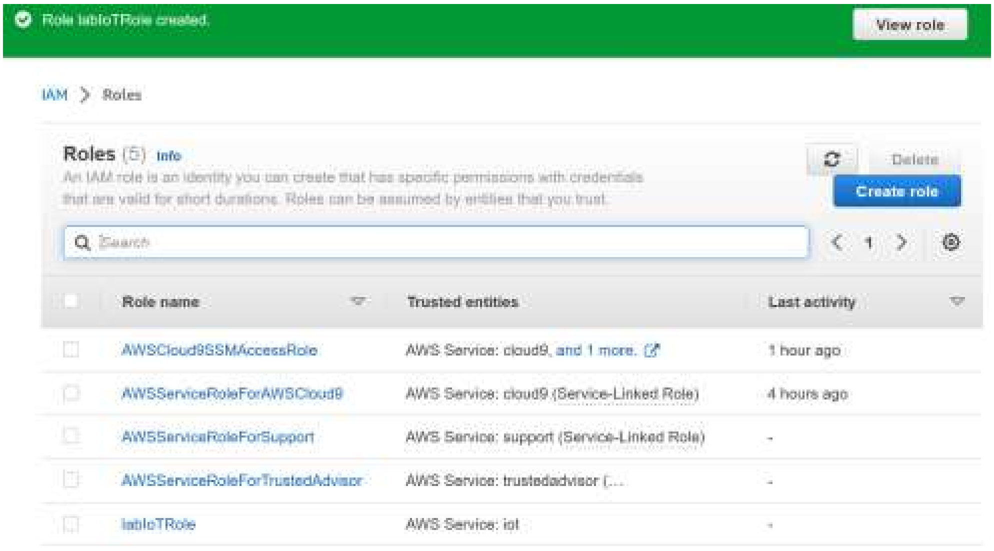
Add tag

You can add up to 50 more tags.

125

Create an IAM Role

- You have now created the IAM Role that will be used in the next section.



Role labIoTRole created.

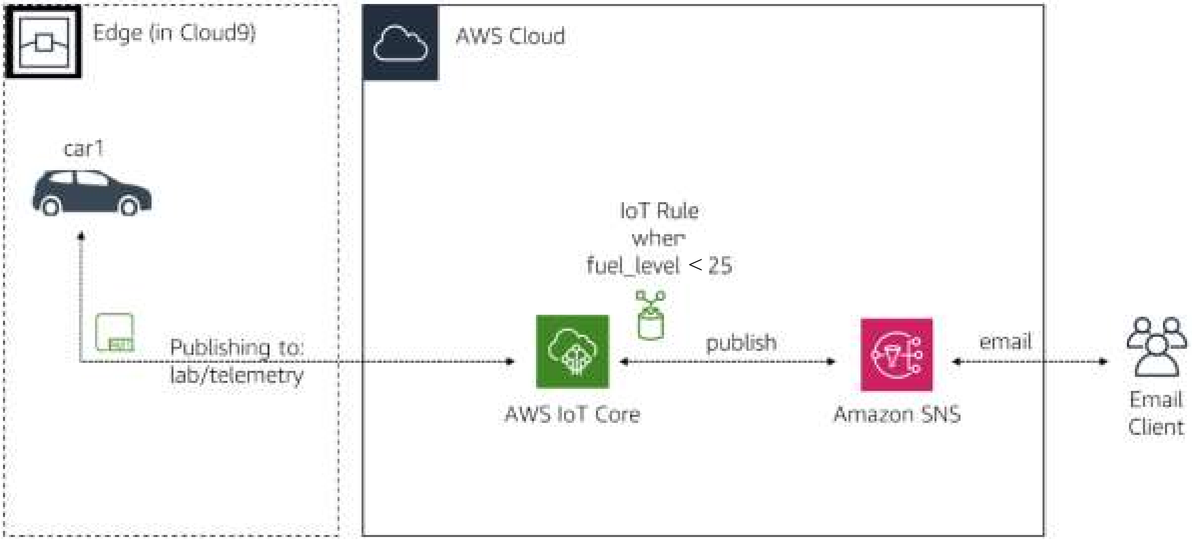
Roles (5) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

□	Role name	Trusted entities	Last activity
□	AWSCloud9SSMAccessRole	AWS Service: cloud9, and 1 more.	1 hour ago
□	AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked Role)	4 hours ago
□	AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)	-
□	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (...	-
□	labIoTRole	AWS Service: iot	-

126

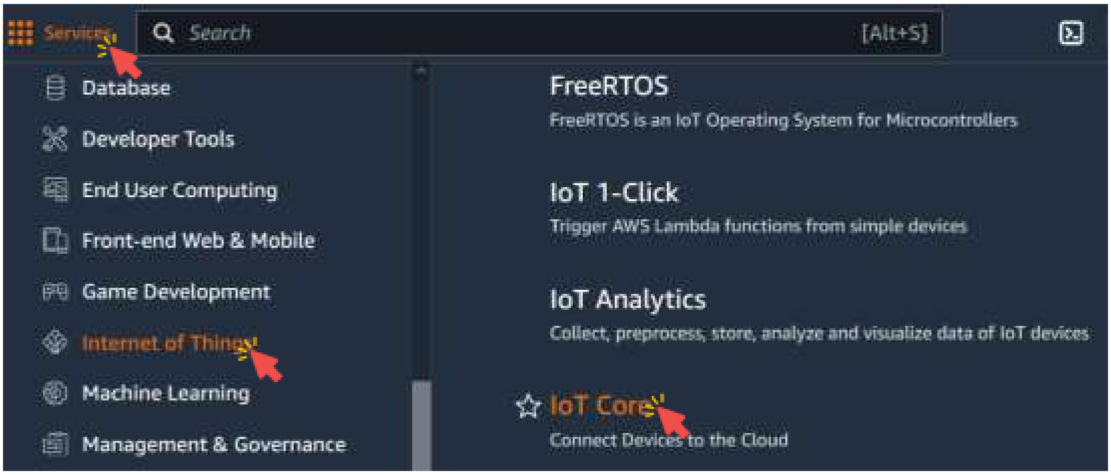
Create an IoT Rule



127

Create an IoT Rule

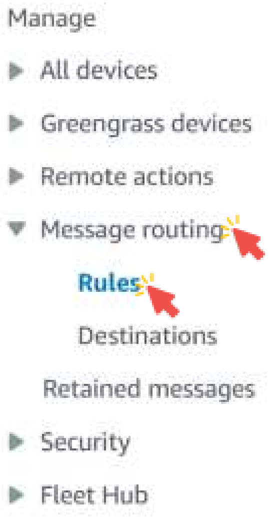
1. In the AWS Management Console, click Services, click Internet of Things and click IoT Core to open the IoT Console.


Services
Search
[Alt+S]
Database
Developer Tools
End User Computing
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
FreeRTOS
FreeRTOS is an IoT Operating System for Microcontrollers
IoT 1-Click
Trigger AWS Lambda functions from simple devices
IoT Analytics
Collect, preprocess, store, analyze and visualize data of IoT devices
IoT Core
Connect Devices to the Cloud

128

Create an IoT Rule

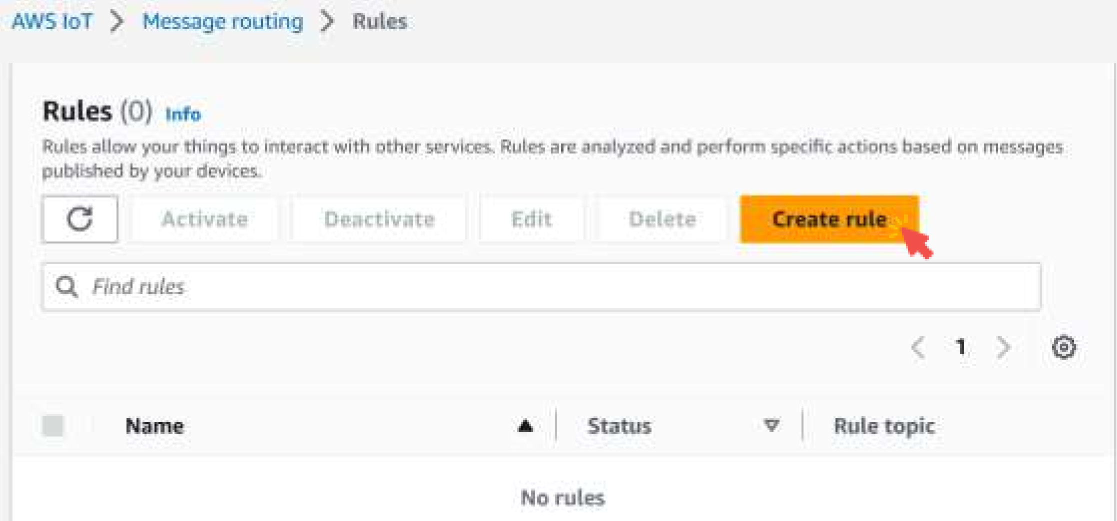
2. Expand Message routing in the left menu, and click Rules.



129

Create an IoT Rule

3. Click Create rule.



Rules (0) Info

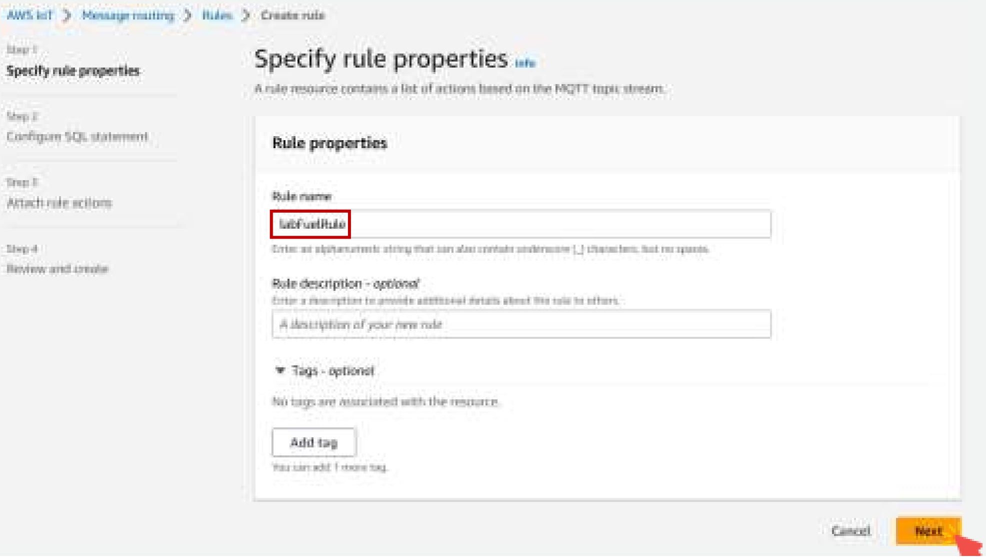
Rules allow your things to interact with other services. Rules are analyzed and perform specific actions based on messages published by your devices.

Name	Status	Rule topic
No rules		

130

Create an IoT Rule

4. For Name, enter labFuelRule, and click Next.



AWS IoT > Message routing > Rules > Create rule

Step 1
Specify rule properties

Step 2
Configure SQL statement

Step 3
Attach rule actions

Step 4
Review and create

Specify rule properties Info

A rule resource contains a list of actions based on the MQTT topic stream.

Rule properties

Rule name

labFuelRule

Enter an alphanumeric string that can also contain underscore (_) characters, but no spaces.

Rule description - *optional*

Enter a description to provide additional details about this rule to others.

A description of your new rule

▼ Tags - optional

No tags are associated with the resource.

You can add 1 more tag.

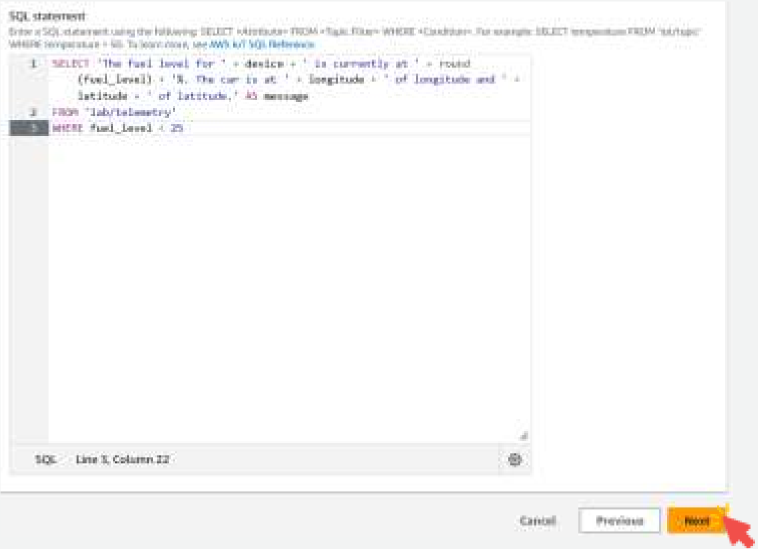
Create an IoT Rule

5. In Configure SQL statement, replace everything with the following:

```
SELECT 'The fuel level for ' + device + ' is currently
at ' + round(fuel_level) + '%. The car is at ' +
longitude + ' of longitude and ' + latitude + ' of
latitude.' AS message
FROM 'lab/telemetry'
WHERE fuel_level < 25
```

Create an IoT Rule

6. Click Next.



SQL statement

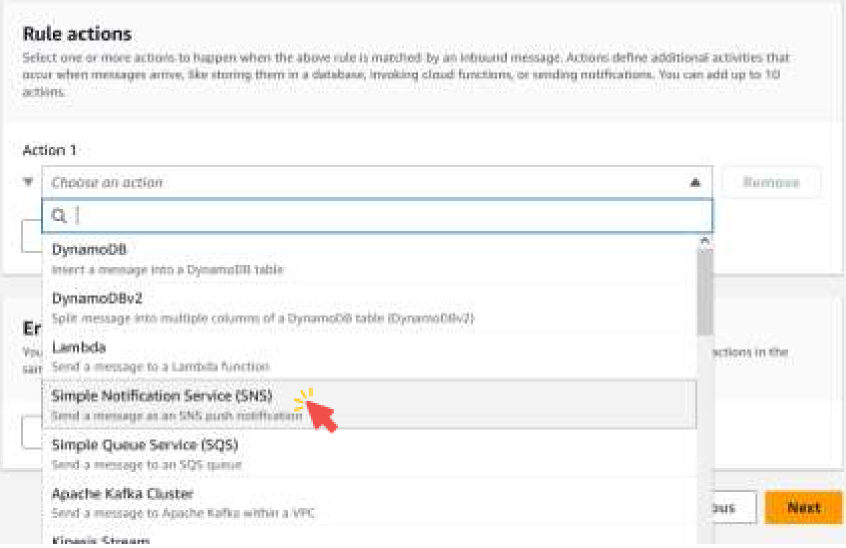
Enter a SQL statement using the following: SELECT <Attribute> FROM <Topic Filter> WHERE <Condition>. For example: SELECT temperature FROM 'iot/topic' WHERE temperature > 50. To learn more, see AWS IoT SQL Reference.

```
SELECT 'The fuel level for ' + device + ' is currently at ' + round
    (fuel_level) + '%. The car is at ' + longitude + ' of longitude and ' +
    latitude + ' of latitude.' AS message
FROM 'lab/telemetry'
WHERE fuel_level < 25
```

133

Create an IoT Rule

7. In Rule actions, select Simple Notification Service (SNS).



Rule 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. You can add up to 10 actions.

Action 1

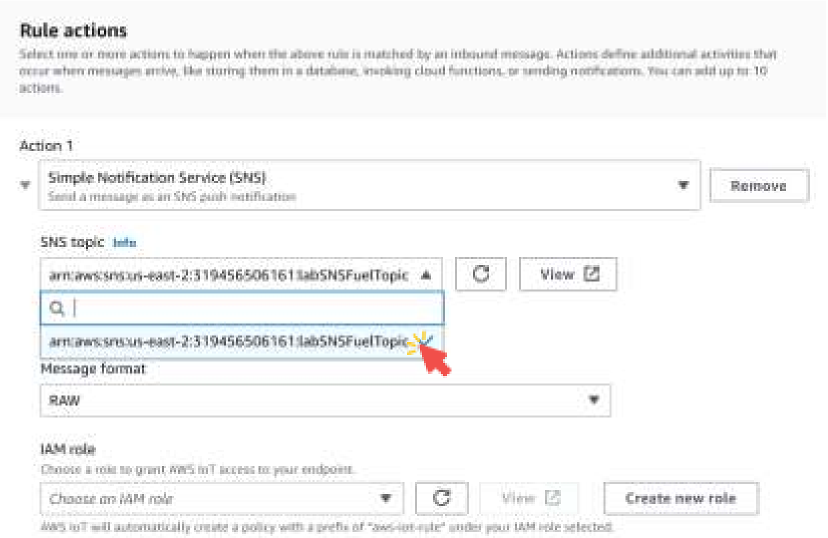
Choose an action

- DynamoDB
  Insert a message into a DynamoDB table
- DynamoDBv2
  Split message into multiple columns of a DynamoDB table (DynamoDBv2)
- Lambda
  Send a message to a Lambda function
- Simple Notification Service (SNS)
  Send a message as an SNS push notification
- Simple Queue Service (SQS)
  Send a message to an SQS queue
- Apache Kafka Cluster
  Send a message to Apache Kafka within a VPC
- Kinesis Stream

134

Create an IoT Rule

8. In SNS topic, select `labSNSFuelTopic`.



Rule 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. You can add up to 10 actions.

Action 1

Simple Notification Service (SNS)
Send a message as an SNS push notification

SNS topic: arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic

Message format: RAW

IAM role
Choose a role to grant AWS IoT access to your endpoint.

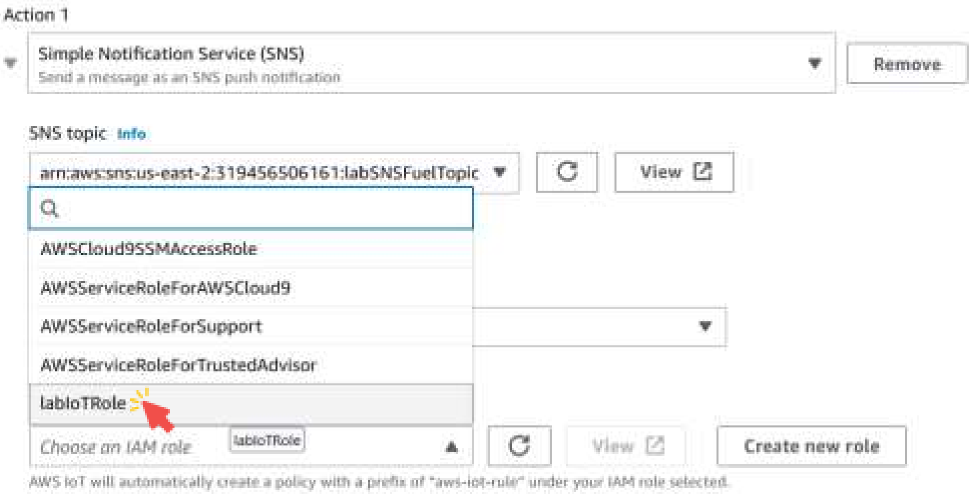
Choose an IAM role

AWS IoT will automatically create a policy with a prefix of "aws-iot-rule" under your IAM role selected.

135

Create an IoT Rule

9. In IAM role, select `labIoTRole`.



Action 1

Simple Notification Service (SNS)
Send a message as an SNS push notification

SNS topic: arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic

AWSCloud9SSMAccessRole
AWSServiceRoleForAWSCloud9
AWSServiceRoleForSupport
AWSServiceRoleForTrustedAdvisor
labIoTRole

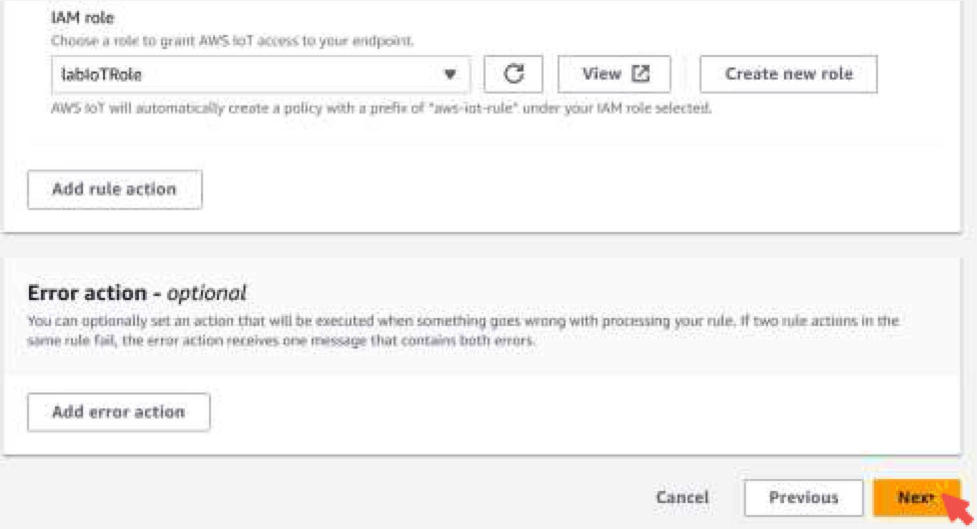
Choose an IAM role

AWS IoT will automatically create a policy with a prefix of "aws-iot-rule" under your IAM role selected.

136

Create an IoT Rule

10. Click Next.



IAM role

Choose a role to grant AWS IoT access to your endpoint.

labIoTRole

AWS IoT will automatically create a policy with a prefix of "aws-iot-rule" under your IAM role selected.

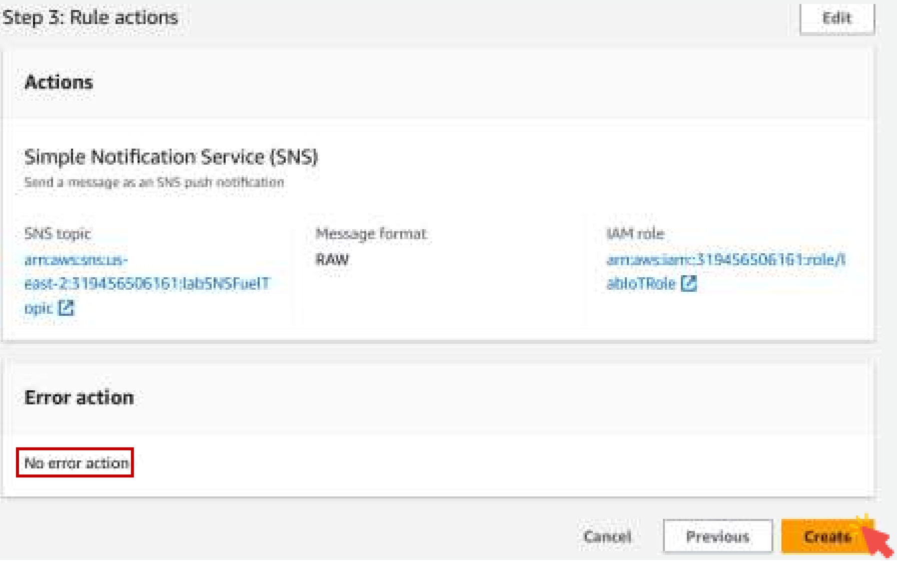
Error action - *optional*

You can optionally set an action that will be executed when something goes wrong with processing your rule. If two rule actions in the same rule fail, the error action receives one message that contains both errors.

137

Create an IoT Rule

11. Click Create.



Step 3: Rule actions

Actions

Simple Notification Service (SNS)

Send a message as an SNS push notification

SNS topic: arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic

Message format: RAW

IAM role: arn:aws:iam::319456506161:role/labIoTRole

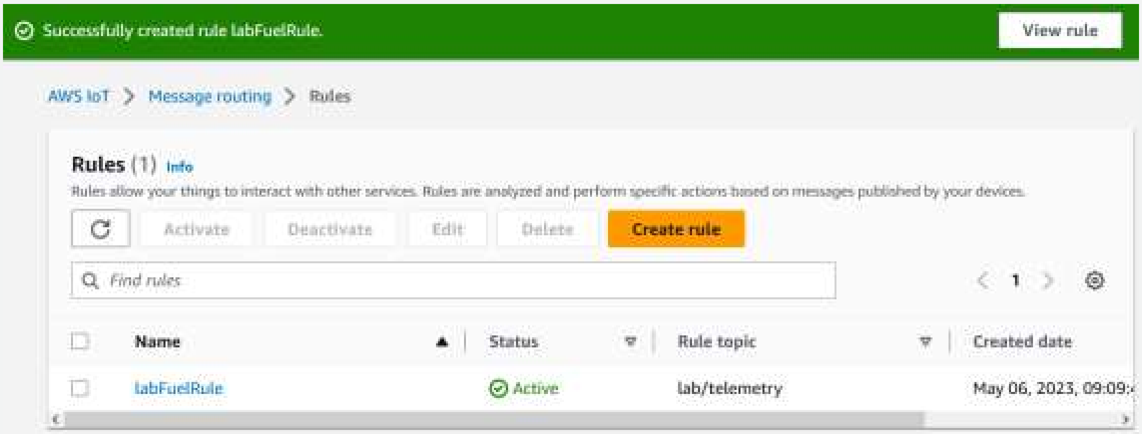
Error action

No error action

138

Create an IoT Rule

- You have now created an IoT Rule that is waiting for the `fuel_level` to be under 25 to send a notification.



Successfully created rule labFuelRule.

View rule

AWS IoT > Message routing > Rules

Rules (1) Info

Rules allow your things to interact with other services. Rules are analyzed and perform specific actions based on messages published by your devices.

Activate | Deactivate | Edit | Delete | Create rule

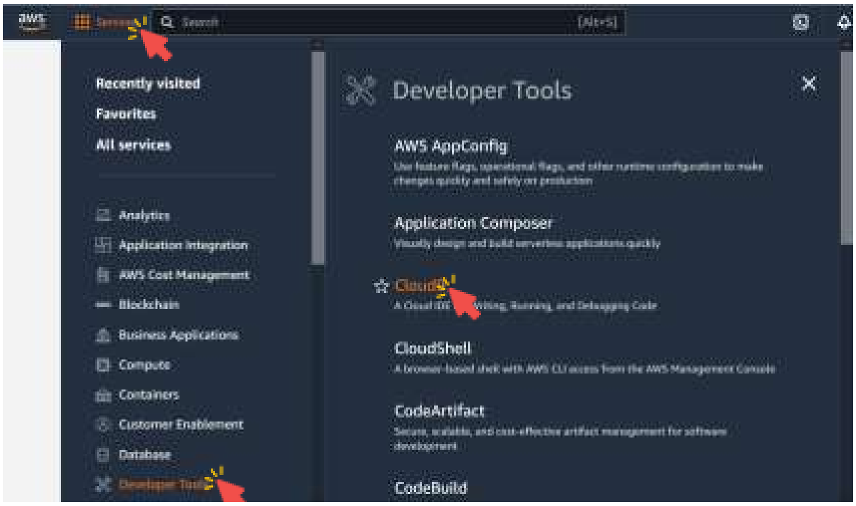
Find rules

□	Name	Status	Rule topic	Created date
□	labFuelRule	✓ Active	lab/telemetry	May 06, 2023, 09:09:4

139

Start Cloud9

1. In the AWS Management Console, click Services, click Developer Tools, and then click Cloud9 to open the Cloud9 dashboard.

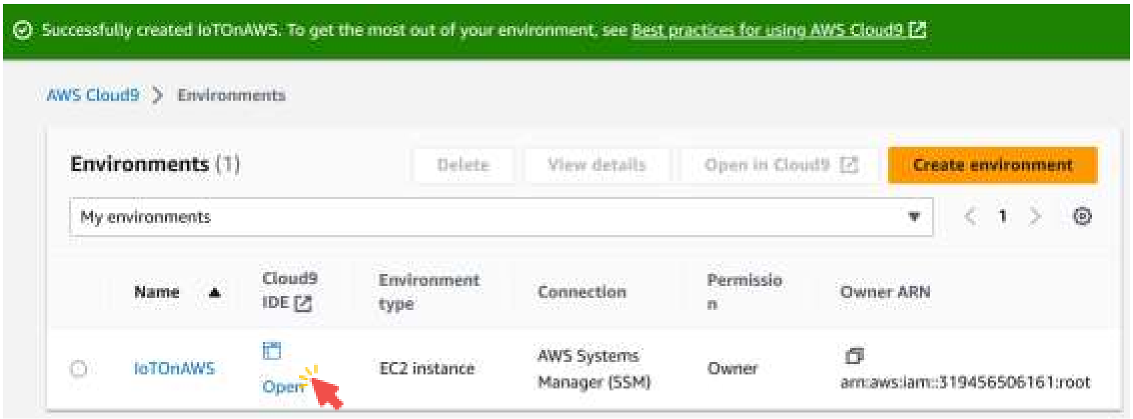


aws
Services
Search
[Alt+S]
Recently visited
Favorites
All services
Analytics
Application Integration
AWS Cost Management
Blockchain
Business Applications
Compute
Containers
Customer Enablement
Database
Developer Tools
Developer Tools
AWS AppConfig
Use feature flags, operational flags, and other runtime configuration to make changes quickly and safely on production
Application Composer
Visually design and build serverless applications quickly
Cloud9
A Cloud IDE for Writing, Running, and Debugging Code
CloudShell
A browser-based shell with AWS CLI access from the AWS Management Console
CodeArtifact
Secure, scalable, and cost-effective artifact management for software development
CodeBuild

140

Start Cloud9

2. Click Open to launch your environment.



Successfully created IoTOnAWS. To get the most out of your environment, see Best practices for using AWS Cloud9

AWS Cloud9 > Environments

Environments (1)

	Name	Cloud9 IDE	Environment type	Connection	Permission	Owner ARN
○	IoTOnAWS	Open	EC2 instance	AWS Systems Manager (SSM)	Owner	arn:aws:iam::319456506161:root

141

Start Cloud9

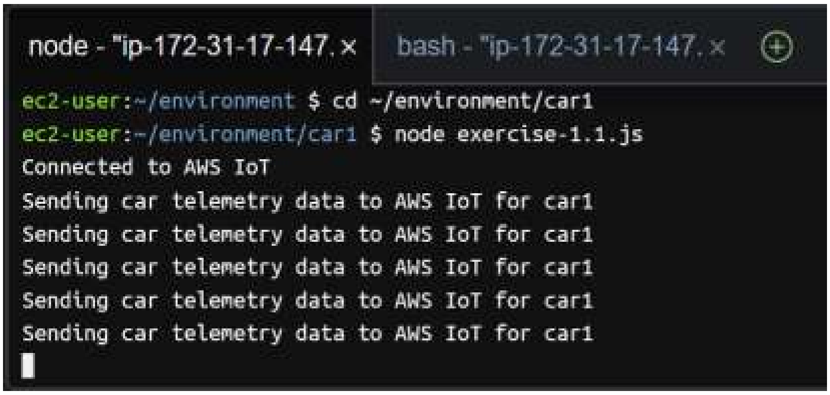
Press Alt-` to go to the next IDE tab.

142

Start Cloud9

3. In the left terminal, execute the commands to start the code for `car1`.

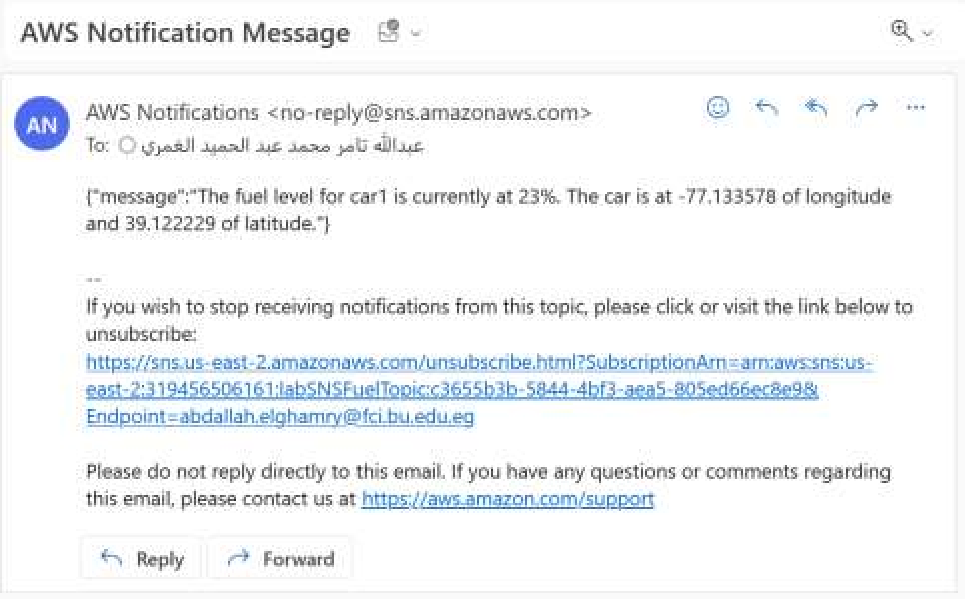
```
cd ~/environment/car1
node exercise-1.1.js
```



```
ec2-user:~/environment $ cd ~/environment/car1
ec2-user:~/environment/car1 $ node exercise-1.1.js
Connected to AWS IoT
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
█
```

143

Start Cloud9



AWS Notification Message

AN

AWS Notifications <no-reply@sns.amazonaws.com>

To: ○ عبدالله تامر محمد عبد الحميد الغمري

{"message":"The fuel level for car1 is currently at 23%. The car is at -77.133578 of longitude and 39.122229 of latitude."}

--

If you wish to stop receiving notifications from this topic, please click or visit the link below to unsubscribe:
https://sns.us-east-2.amazonaws.com/unsubscribe.html?SubscriptionArn=arn:aws:sns:us-east-2:319456506161:labSNSFuelTopic:c3655b3b-5844-4bf3-aea5-805ed66ec8e9&Endpoint=abdallah.elghamry@fci.bu.edu.eg

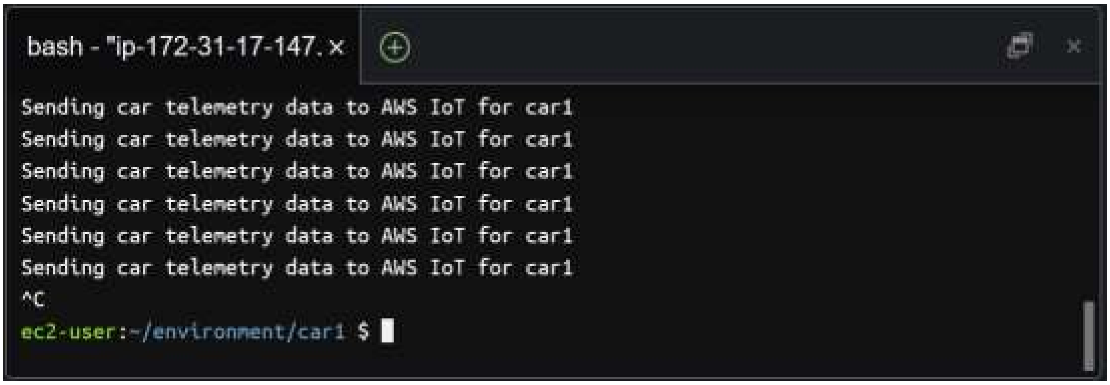
Please do not reply directly to this email. If you have any questions or comments regarding this email, please contact us at https://aws.amazon.com/support

144

Delete The Created Resources

1. Stop car1.

Press Ctrl + C in the Cloud9 terminal to stop the car from interacting with AWS IoT.

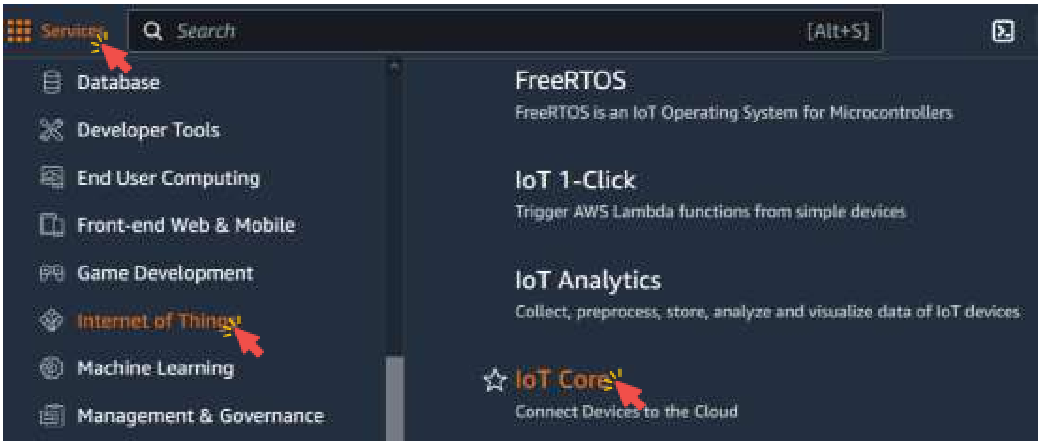


```
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
Sending car telemetry data to AWS IoT for car1
^C
ec2-user:~/environment/car1 $
```

Delete The Created Resources

2. Delete IoT Rule.

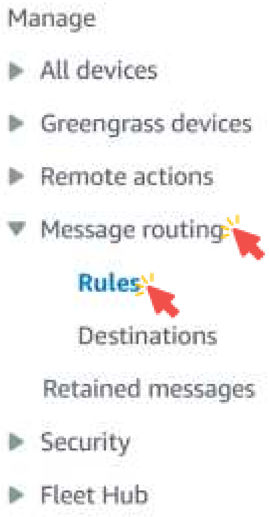
2.1 In the AWS Management Console, click Services, click Internet of Things and click IoT Core to open the IoT Console.



Services
Search
[Alt+S]
Database
Developer Tools
End User Computing
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
FreeRTOS
FreeRTOS is an IoT Operating System for Microcontrollers
IoT 1-Click
Trigger AWS Lambda functions from simple devices
IoT Analytics
Collect, preprocess, store, analyze and visualize data of IoT devices
IoT Core
Connect Devices to the Cloud

Delete The Created Resources

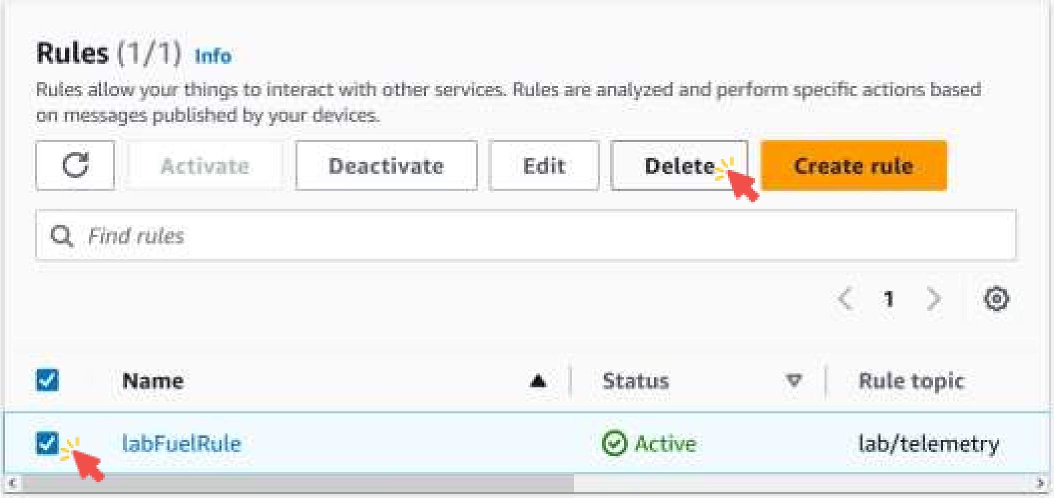
2.2 Expand Message routing in the left menu, and click Rules.



147

Delete The Created Resources

2.3 Select `labFuelRule`, and click Delete.



Rules allow your things to interact with other services. Rules are analyzed and perform specific actions based on messages published by your devices.

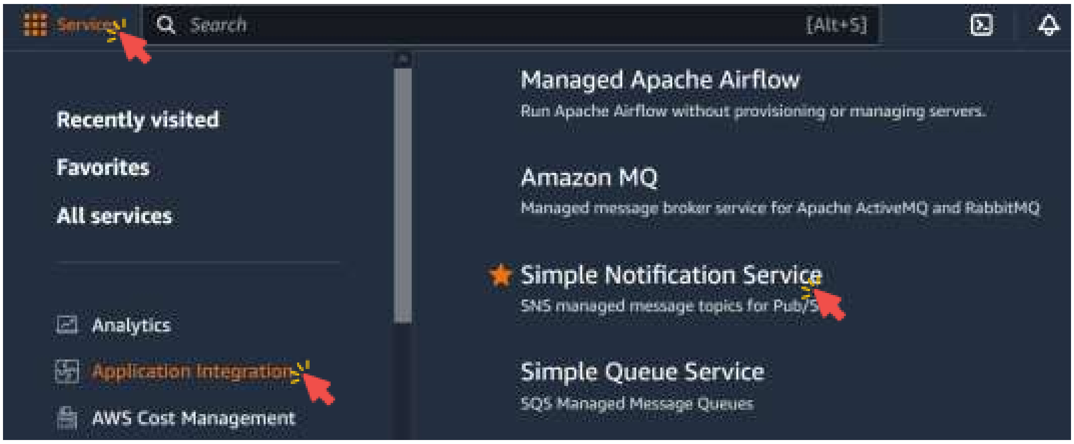
☑	Name	Status	Rule topic
☑	labFuelRule	Active	lab/telemetry

148

Delete The Created Resources

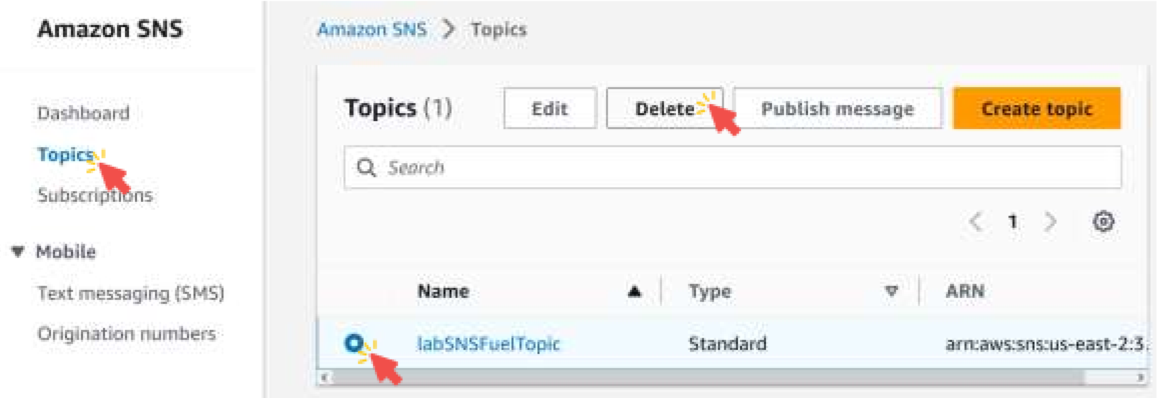
3. Delete SNS Topic.

3.1 In the AWS Management Console, click Services, click Application Integration, and then click Simple Notification Service to go to the SNS.


Services
Search
[Alt+S]
Recently visited
Favorites
All services
Analytics
Application Integration
AWS Cost Management
Managed Apache Airflow
Run Apache Airflow without provisioning or managing servers.
Amazon MQ
Managed message broker service for Apache ActiveMQ and RabbitMQ
Simple Notification Service
SNS managed message topics for Pub/S
Simple Queue Service
SQS Managed Message Queues

Delete The Created Resources

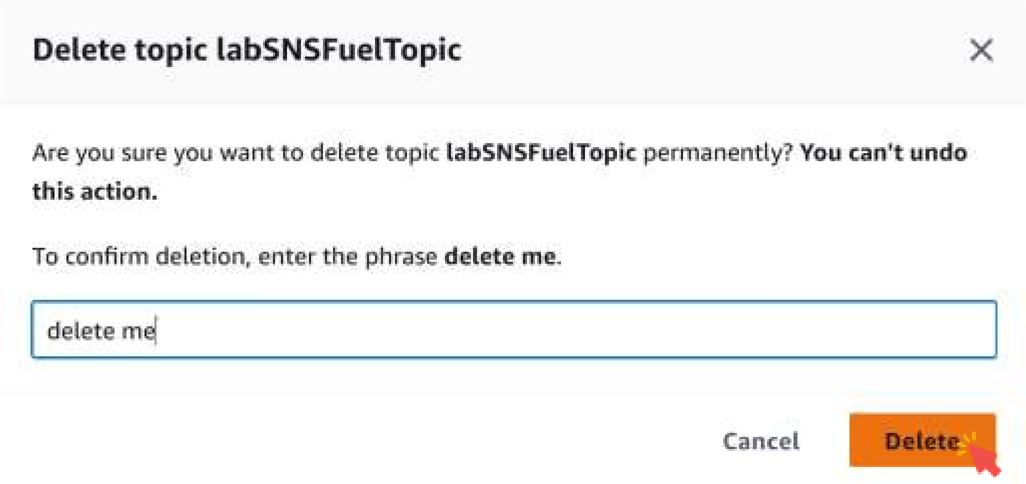
3.2 Click on Topics, select `labSNSFuelTopic`, and click Delete.



	Name ▲	Type ▽	ARN
●	labSNSFuelTopic	Standard	arn:aws:sns:us-east-2:3

Delete The Created Resources

3.3 Click Delete.



Delete topic labSNSFuelTopic ✕

Are you sure you want to delete topic **labSNSFuelTopic** permanently? **You can't undo this action.**

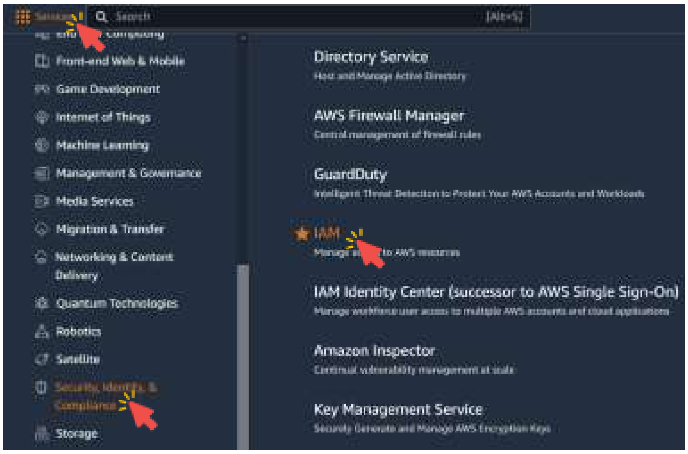
To confirm deletion, enter the phrase **delete me**.

delete me

Delete The Created Resources

4. Delete IAM Role.

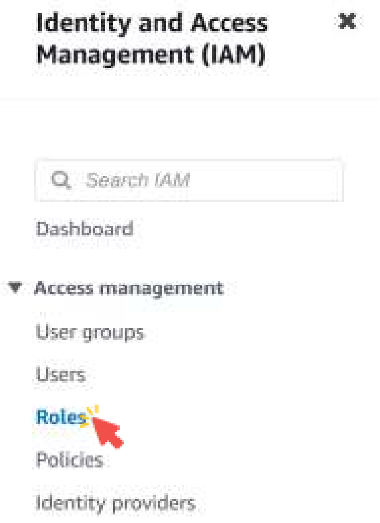
4.1 In the AWS Management Console, click Services, click Security, Identity, & Compliance, and then click IAM to go to the IAM dashboard.



Services
Search
[Alt+S]
Front-end Web & Mobile
Game Development
Internet of Things
Machine Learning
Management & Governance
Media Services
Migration & Transfer
Networking & Content Delivery
Quantum Technologies
Robotics
Satellite
Security, Identity, & Compliance
Storage
Directory Service
Host and Manage Active Directory
AWS Firewall Manager
Central management of firewall rules
GuardDuty
Intelligent Threat Detection to Protect Your AWS Accounts and Workloads
IAM
Manage a to AWS resources
IAM Identity Center (successor to AWS Single Sign-On)
Manage workforce user access to multiple AWS accounts and cloud applications
Amazon Inspector
Continual vulnerability management at scale
Key Management Service
Securely Generate and Manage AWS Encryption Keys

Delete The Created Resources

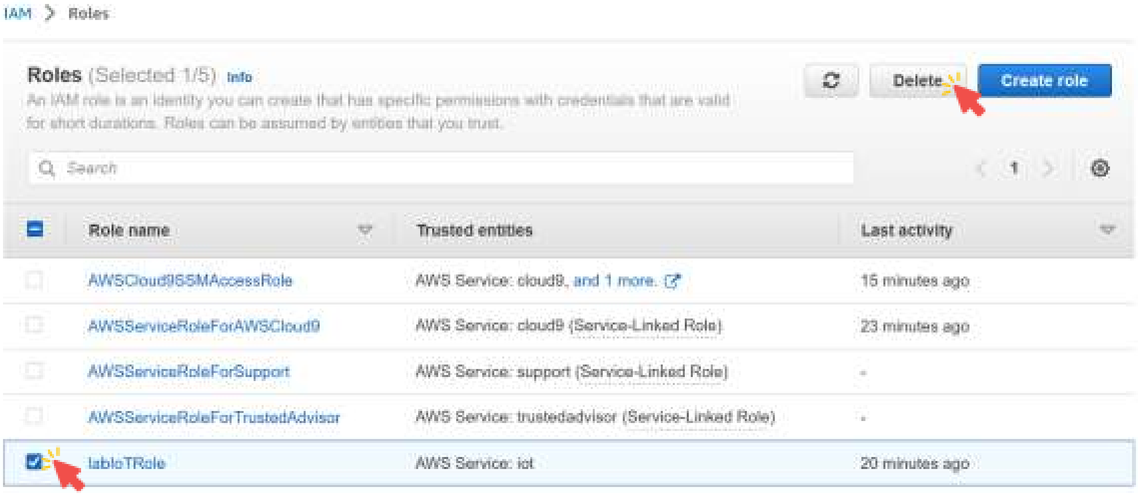
4.2 In the left navigation menu, click Roles.



153

Delete The Created Resources

4.3 Select labIoTRole, and click Delete.



An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

☐	Role name	Trusted entities	Last activity
☐	AWSCloud9SSMAccessRole	AWS Service: cloud9, and 1 more.	15 minutes ago
☐	AWSServiceRoleForAWSCloud9	AWS Service: cloud9 (Service-Linked Role)	23 minutes ago
☐	AWSServiceRoleForSupport	AWS Service: support (Service-Linked Role)	-
☐	AWSServiceRoleForTrustedAdvisor	AWS Service: trustedadvisor (Service-Linked Role)	-
☑	labIoTRole	AWS Service: iot	20 minutes ago

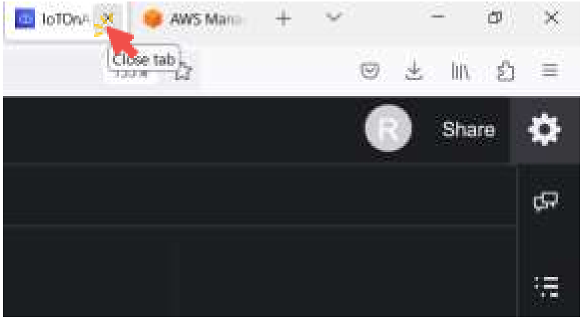
154

Delete The Created Resources

5. Stop the Cloud9 environment.
   - The Cloud9 will automatically shutdown after 30 minutes of inactivity.
   - For your Cloud9 environment to be considered inactive, you need to close the browser tab.
   - All the settings will be saved.

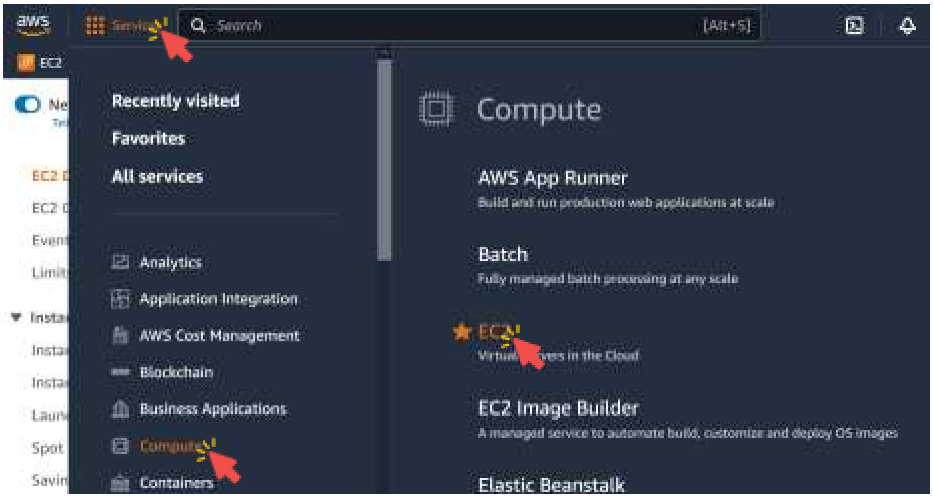
Delete The Created Resources

5.1 Close the browser tab where your environment was running.



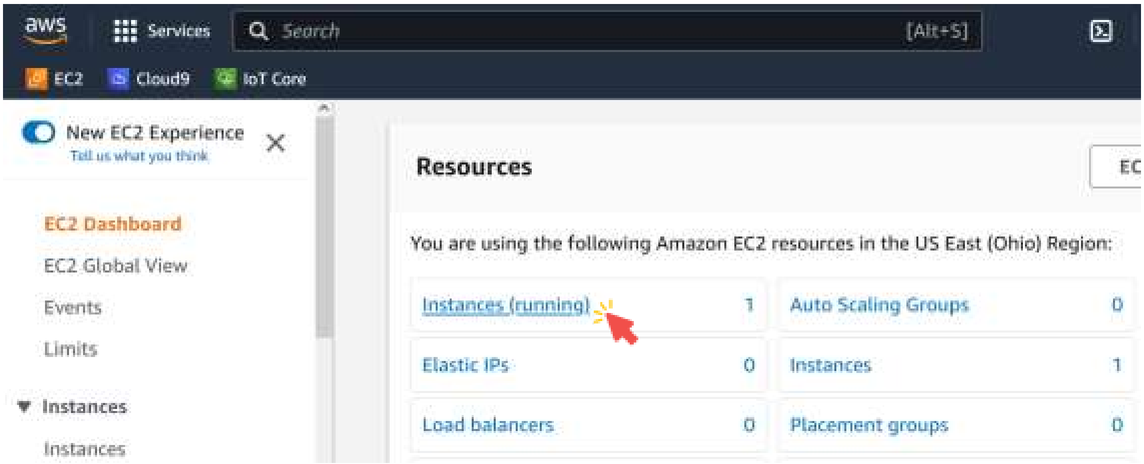
Delete The Created Resources

5.2 In the AWS Management Console, click Services, click Compute, and then click EC2 to open the EC2 console.



Delete The Created Resources

5.3 Click Instances in the left menu.



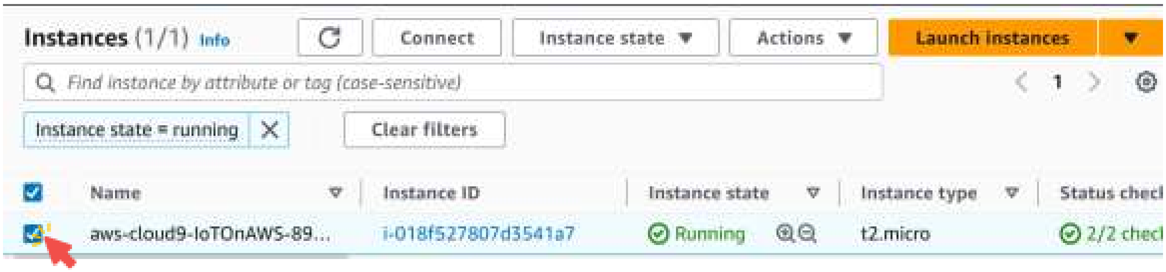
Resources

You are using the following Amazon EC2 resources in the US East (Ohio) Region:

Instances (running)	1	Auto Scaling Groups	0
Elastic IPs	0	Instances	1
Load balancers	0	Placement groups	0

Delete The Created Resources

5.4 Select the EC2 Instance that starts with aws-cloud9-IoTOnAWS.

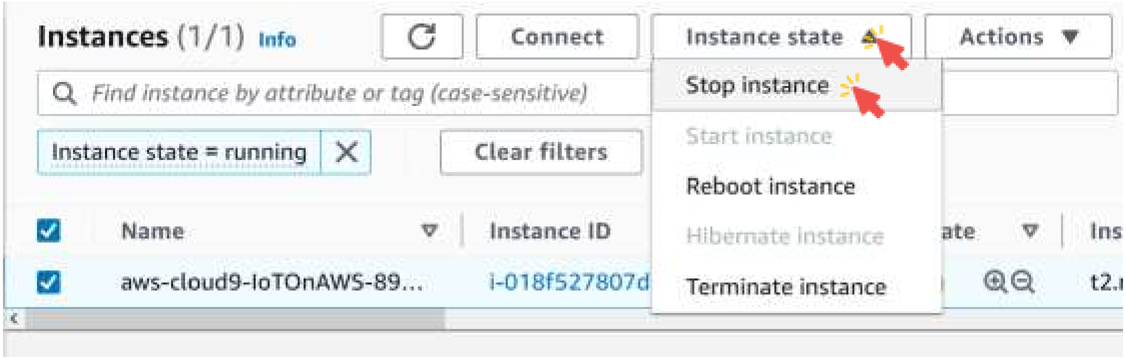


☑	Name	Instance ID	Instance state	Instance type	Status checl
☑	aws-cloud9-IoTOnAWS-89...	i-018f527807d3541a7	Running	t2.micro	2/2 checl

159

Delete The Created Resources

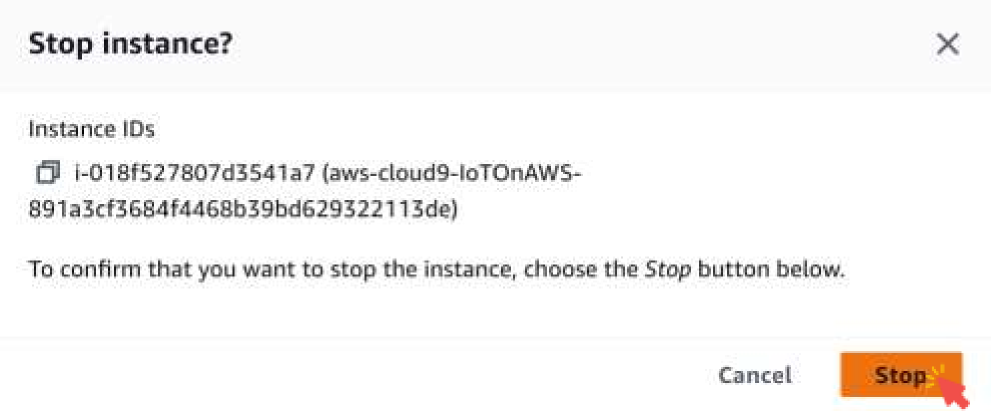
5.5 Click Instance State → Stop instance.



160

Delete The Created Resources

5.6 Click Stop.



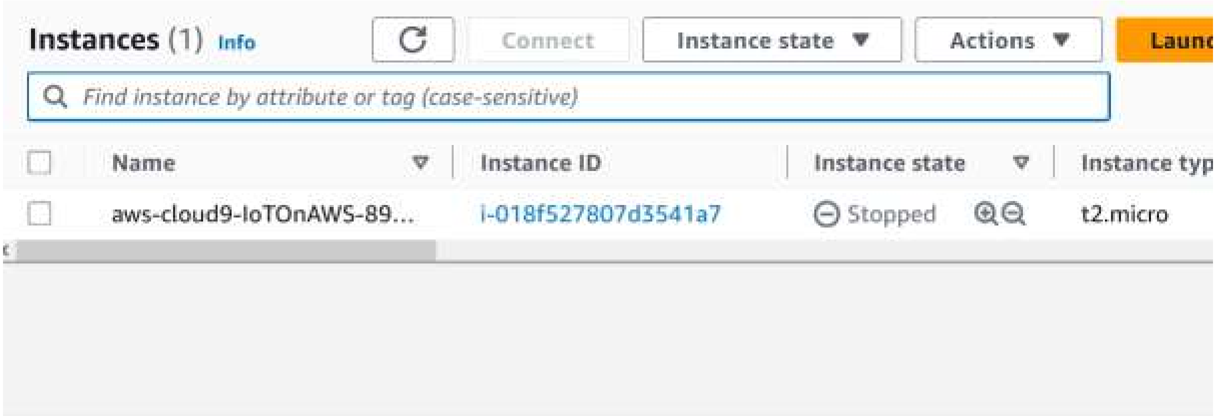
Stop instance?

Instance IDs

i-018f527807d3541a7 (aws-cloud9-IoTOnAWS-891a3cf3684f4468b39bd629322113de)

To confirm that you want to stop the instance, choose the *Stop* button below.

Delete The Created Resources



Instances (1) Info

□	Name	Instance ID	Instance state	Instance typ
□	aws-cloud9-IoTOnAWS-89...	i-018f527807d3541a7	⊖ Stopped	t2.micro