



MQTT

Connecting to the AWS, Amazon Web Services

Vers. 1.1 – Jul 2021

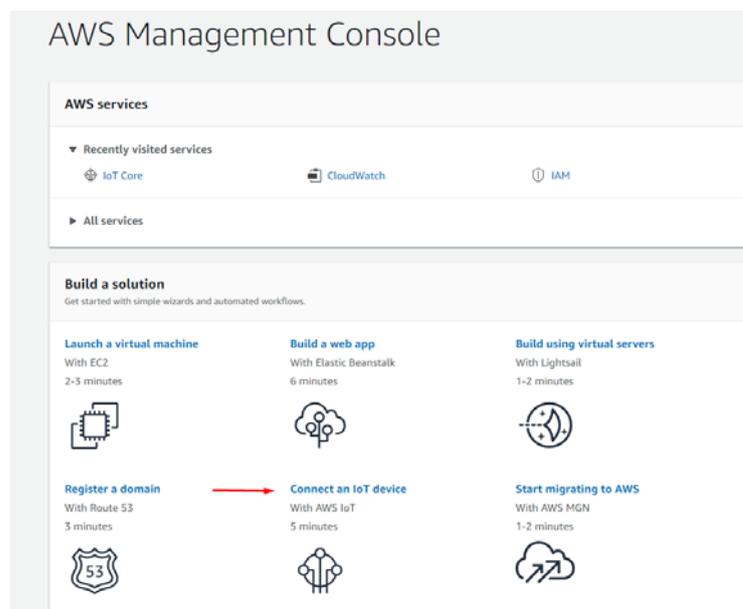
1. Introduction

All of Infinite's devices that support the MQTT protocol, are capable to connect to any local or remote MQTT Broker. Amazon Web Services is a subsidiary of Amazon providing on-demand cloud computing platforms and APIs to individuals, companies, and governments, on a metered pay-as-you-go basis.

This document is a brief how-to guide for all device communications between Infinite's devices and the AWS which supports MQTT connectivity.

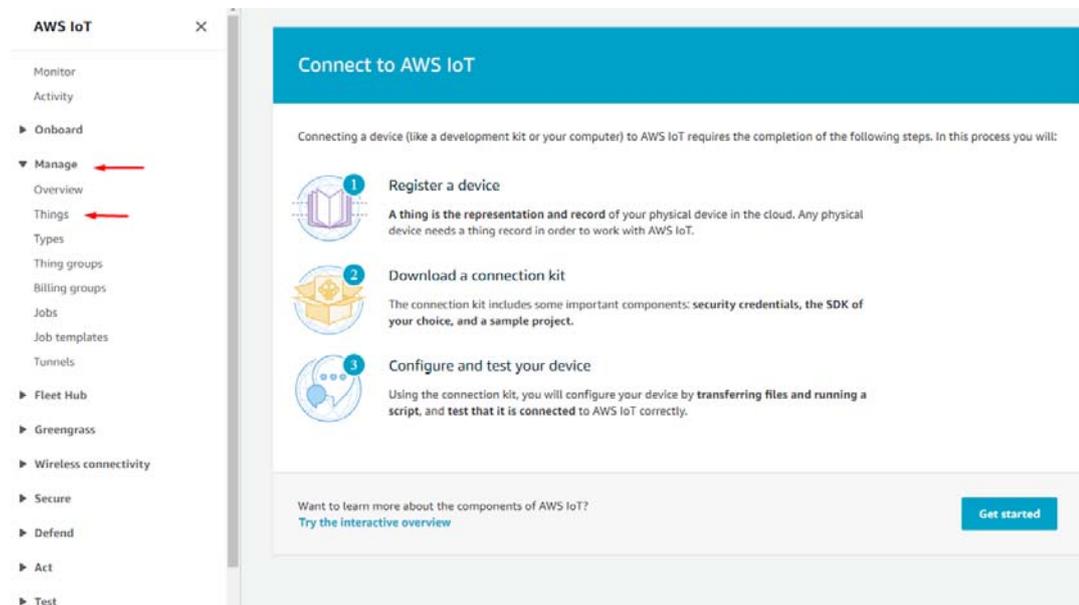
2. AWS Console

After creating an AWS account, navigate to the AWS Management Console page and click Connect an IoT device.

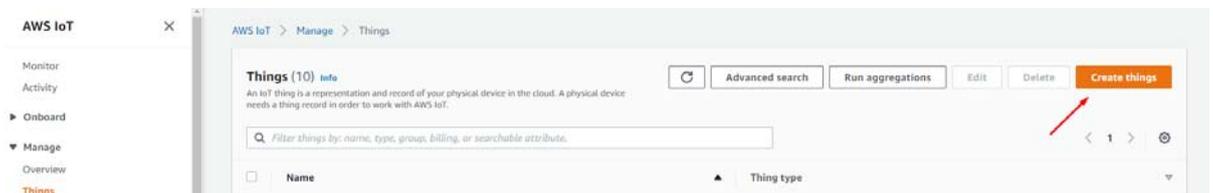


MQTT - Connecting to the AWS, Amazon Web Services

Open the Manage tab and click Things.



Click Create things.



Create a single thing.

AWS IoT > Manage > Things > Create things

Create things [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.

Number of things to create

Create single thing
Create a thing resource to register a device. Provision the certificate and policy necessary to allow the device to connect to AWS IoT.

Create many things
Create a task that creates multiple thing resources to register devices and provision the resources those devices require to connect to AWS IoT.

[Cancel](#) [Next](#)

Give the Thing a name.

Thing properties [Info](#)

Thing name

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

Additional configurations

You can use these configurations to add detail that can help you to organize, manage, and search your things.

- ▶ Thing type - optional
- ▶ Searchable thing attributes - optional
- ▶ Thing groups - optional
- ▶ Billing group - optional

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 HTTP 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.

[Cancel](#) [Next](#)

Auto-generate a new certificate. (AWS requires TLS communications)

Configure device certificate - optional [Info](#)

A device requires a certificate to connect to AWS IoT. You can choose how you to register a certificate for your device now, or you can create and register a certificate for your device later. Your device won't be able to connect to AWS IoT until it has an active certificate with an appropriate policy.

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.

Cancel Previous **Next**

Create a policy to attach to the certificate.

Attach policies to certificate - optional [Info](#)

AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the device certificate applies this access to the device.

Policies (1) ↻ Create policy ↗

Select up to 10 policies to attach to this certificate.

< 1 > ⚙️

<input type="checkbox"/>	Name
<input type="checkbox"/>	nbiot

Cancel Previous **Create thing**

Name the policy and click advanced mode to define the types of actions that can be performed by our device.

Add statements

Policy statements define the types of actions that can be performed by a resource. Basic mode

```
1 [{"Version": "2012-10-17",
2  "Statement": [
3    {
4      "Effect": "",
5      "Action": "*",
6      "Resource": "*"
7    }
8  ]
9 }
10 ]
```

Delete the pre-existing statements and paste the following ones.

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

This policy is for testing purposes (it allows all communications to and from the device) and should be adjusted for your requirements.

Click refresh and choose the policy you just created and click Create thing.

Attach policies to certificate - *optional* Info

AWS IoT policies grant or deny access to AWS IoT resources. Attaching policies to the device certificate applies this access to the device.

Policies (1/2)
Select up to 10 policies to attach to this certificate.

Filter policies

<input type="checkbox"/>	Name
<input type="checkbox"/>	nbiot
<input checked="" type="checkbox"/>	Open_Comm

Cancel Previous **Create thing**

In the windows that pops up you can download the certificates that were created.

Download certificates and keys

Download certificate and key files to install on your device so that it can connect to AWS.

Device certificate
You can activate the certificate now, or later. The certificate must be active for a device to connect to AWS IoT.

Device certificate
3e18ee0b1d5...te.pem.crt **Deactivate certificate** **Download**

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
3e18ee0b1d5b2bbdbbd17c5...512d954-public.pem.key **Download**

Private key file
3e18ee0b1d5b2bbdbbd17c5...12d954-private.pem.key **Download**

Root CA certificates
Download the root CA certificate file that corresponds to the type of data endpoint and cipher suite you're using. You can also download the root CA certificates later.

Amazon trust services endpoint
RSA 2048 bit key: Amazon Root CA 1 **Download**

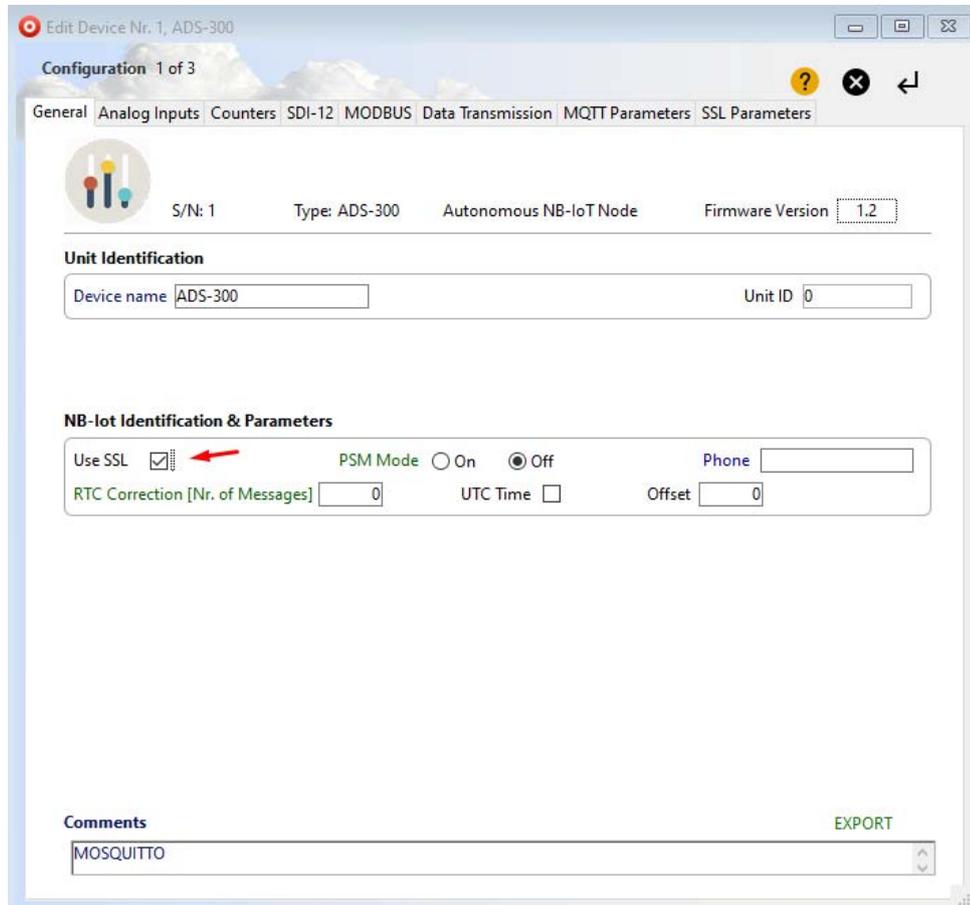
Amazon trust services endpoint
ECC 256 bit key: Amazon Root CA 3 **Download**

If you don't see the root CA certificate that you need here, AWS IoT supports additional root CA certificates. These root CA certificates and others are available in our developer guides. [Learn more](#)

Done

3. Device Configuration with WA Manager

In the Edit Device window in WA Manager, tick the Use SSL box.

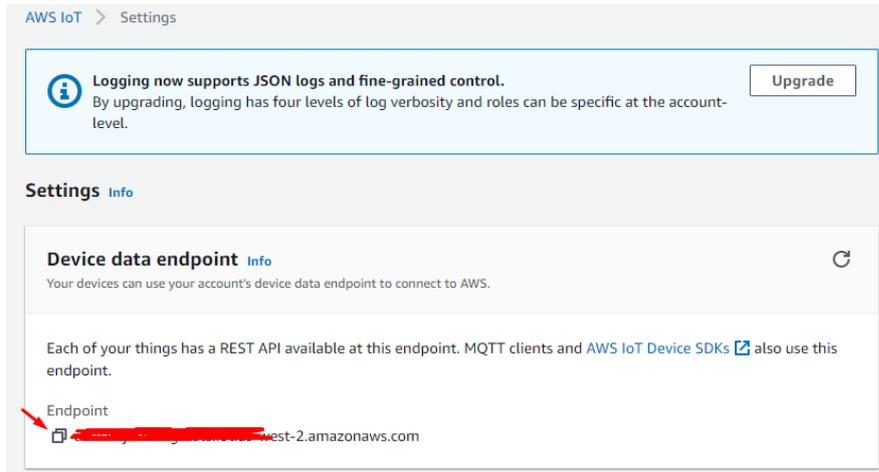


Next, we configure the MQTT parameters.

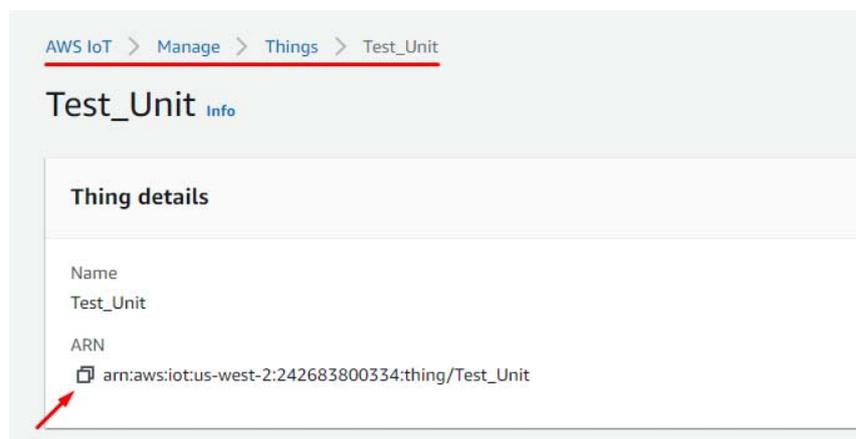
Although AWS supports MQTT connectivity, it is not a pure MQTT Broker and so it has some limitations regarding its MQTT parameters.

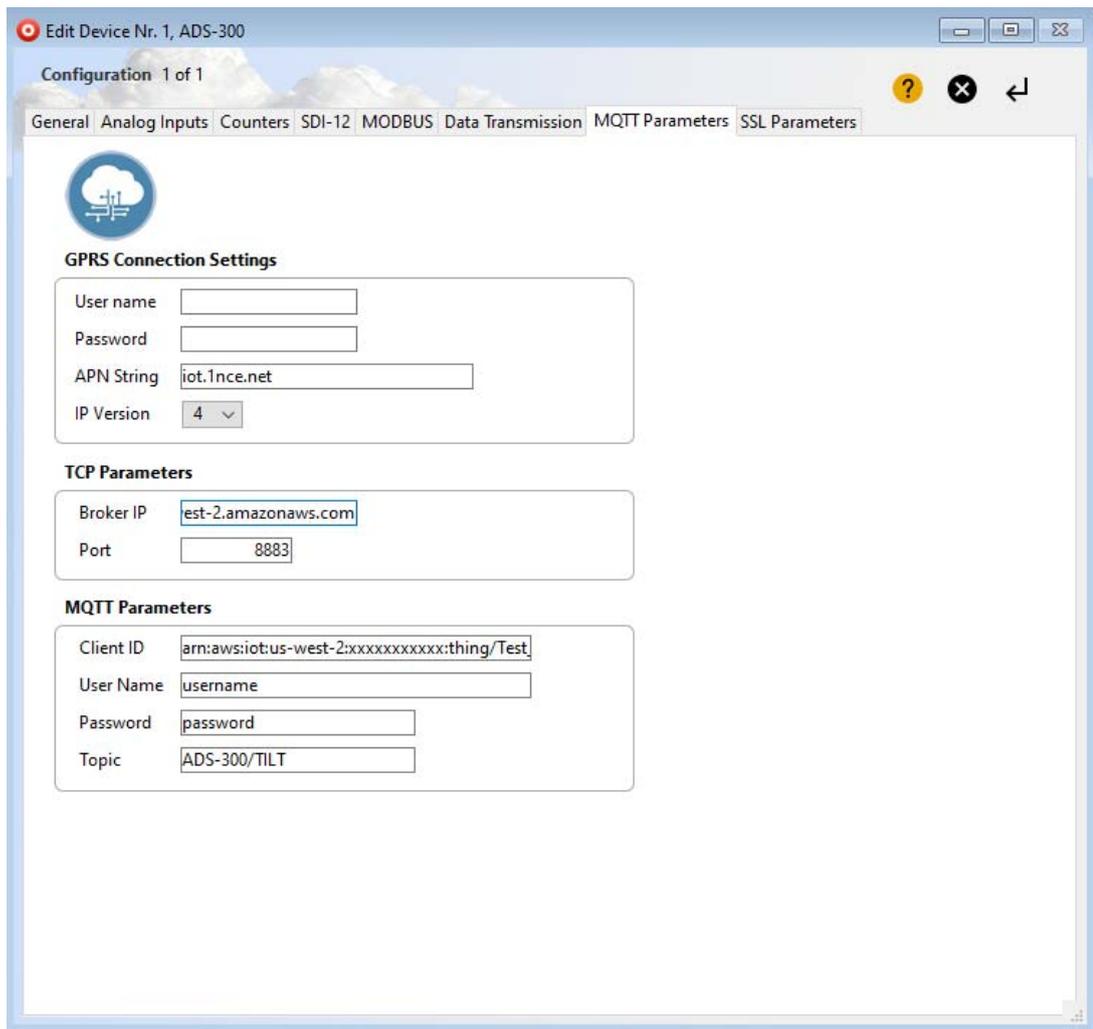
For the Broker IP, the Device data endpoint must be used that can be found in the AWS IoT Settings tab.

MQTT - Connecting to the AWS, Amazon Web Services



For the Client ID, the ARN (Amazon Resource Name) must be used that can be found in the Things tab.

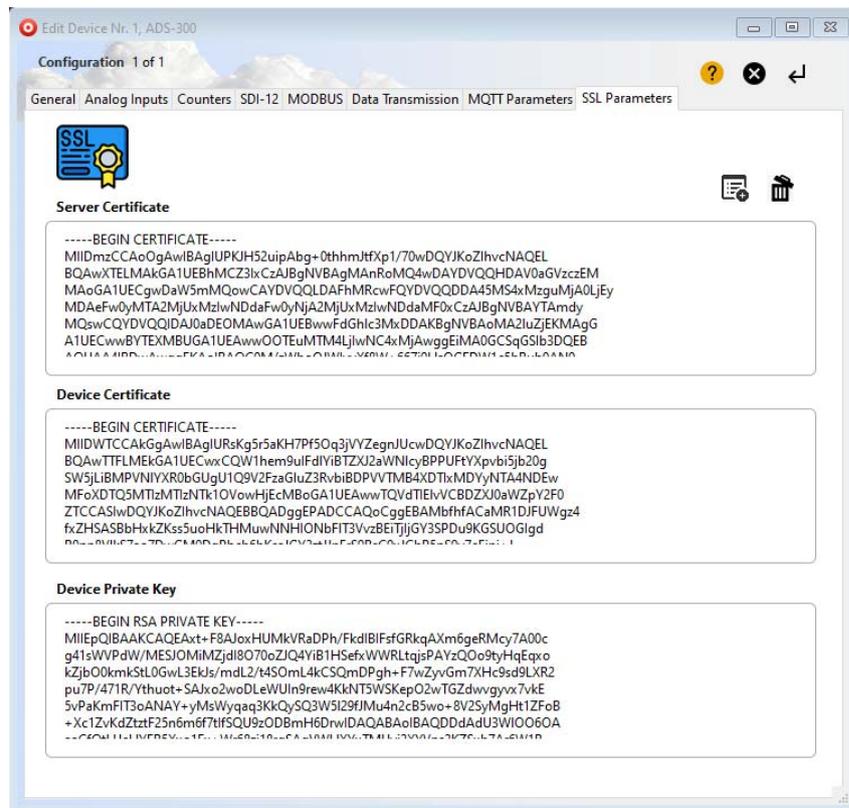




The screenshot shows a software configuration window titled "Edit Device Nr. 1, ADS-300". The window has a tabbed interface with the following tabs: "General", "Analog Inputs", "Counters", "SDI-12", "MODBUS", "Data Transmission", "MQTT Parameters", and "SSL Parameters". The "MQTT Parameters" tab is currently selected. The window contains three main sections of settings:

- GPRS Connection Settings:**
 - User name:
 - Password:
 - APN String:
 - IP Version: (dropdown menu)
- TCP Parameters:**
 - Broker IP:
 - Port:
- MQTT Parameters:**
 - Client ID:
 - User Name:
 - Password:
 - Topic:

Lastly, in the SSL Parameters tab, we copy and paste the three files needed for the TLS communication: Server Certificate (CA), Device Certificate and Device Private Key.



The Server Certificate is the Amazon trust services that you previously downloaded, the Device Certificate is the file you downloaded and the Device Private Key is the private key file. These files should be first opened with Notepad++ and their contents should be copy and pasted in the above tab. All files must be PEM formatted.

Your device can now securely connect to the AWS and send your encrypted telemetry data safely.

Disclaimer:

AWS, Amazon Web services is registered trademark of Amazon.com Inc, USA. All products and software mentioned in this document for educational and demonstration purposes.

Revision: 1.0

© 2021, Infinite Informatics Ltd

Infinite Informatics, Ltd

1, Valaoritou Street
GR-54626 Thessaloniki, Greece
Phone: +30-2310-553545
E: info@indinf.gr
W: www.infinite.com.gr