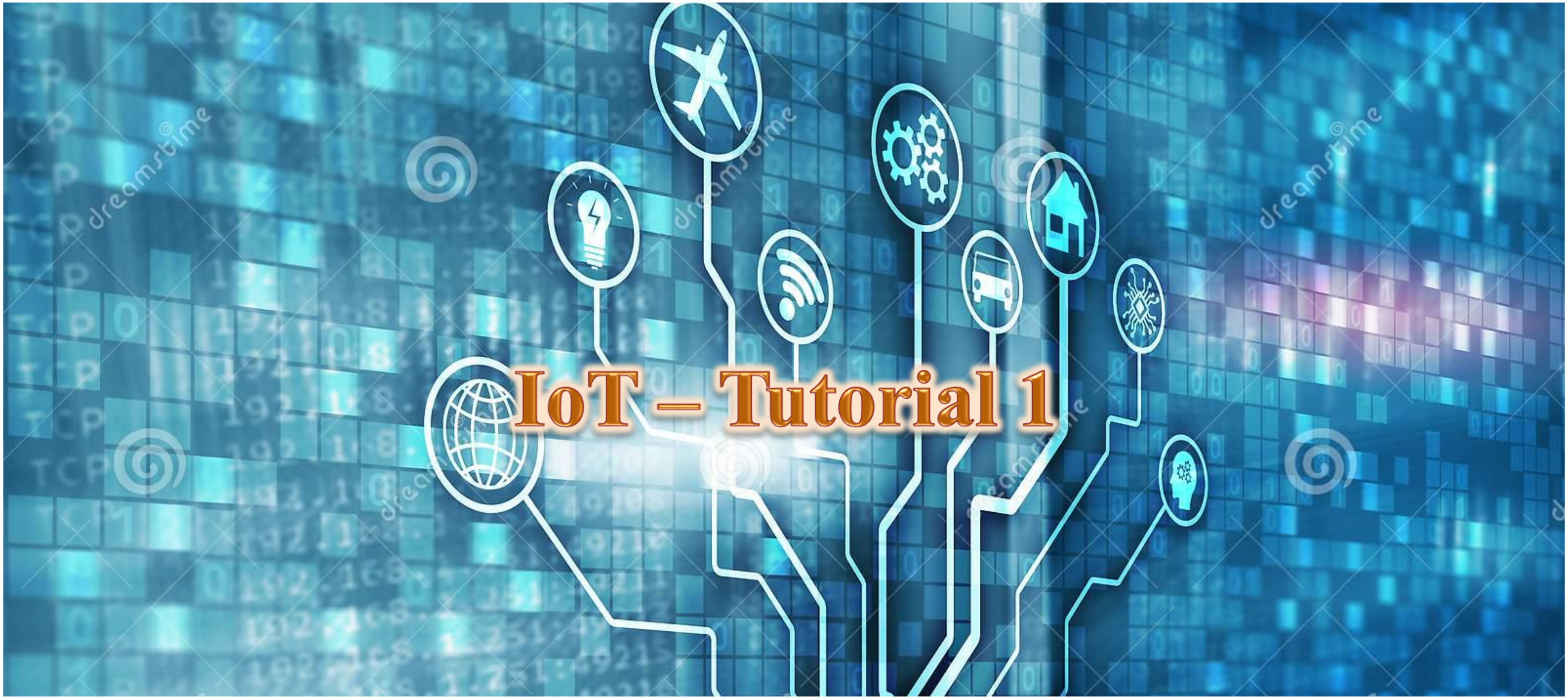


IoT – Tutorial 1

MQTT Protocol

MQTT Org

Mouli Sankaran

Focus: IoT - Tutorial 1

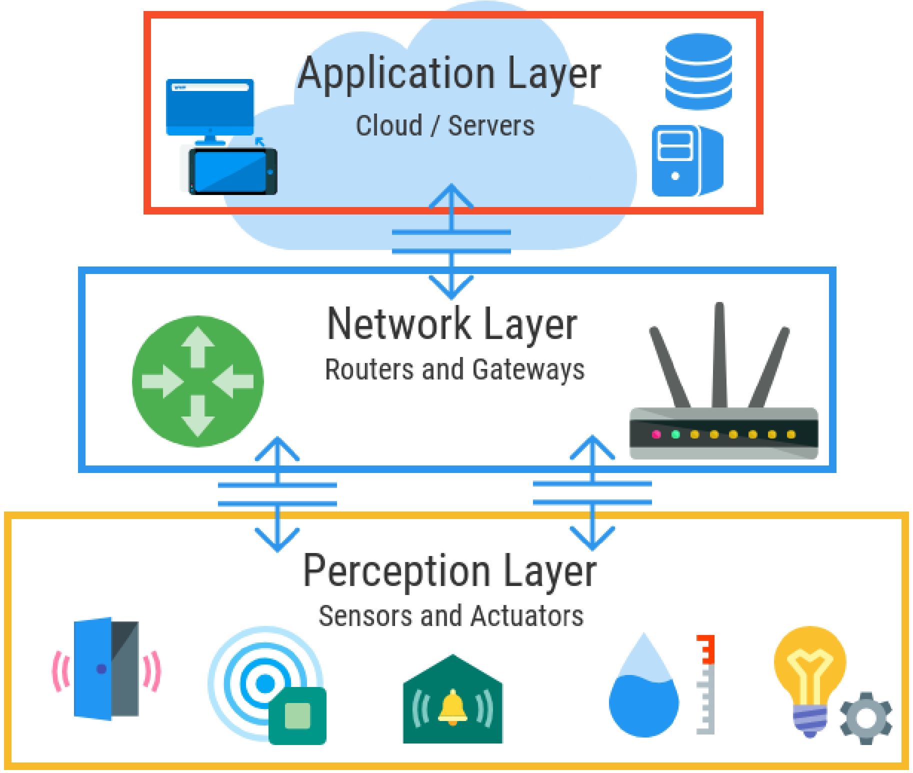
IoT – Tutorial 0 – ESP32 Installation Guide

- MQTT Protocol
  - Introduction
  - Features
- Mosquitto
  - Installation on Windows 10
  - Usage
- MQTT on Cloud - HiveMQ
  - HiveMQ Registration
  - Getting started with HiveMQ
  - Demo

IoT – Tutorial 2 – Porting MQTT Client on to ESP32 Board

IoT Architecture & Communication Requirements

Generic IoT Architecture



Application Layer

Cloud / Servers

Network Layer

Routers and Gateways

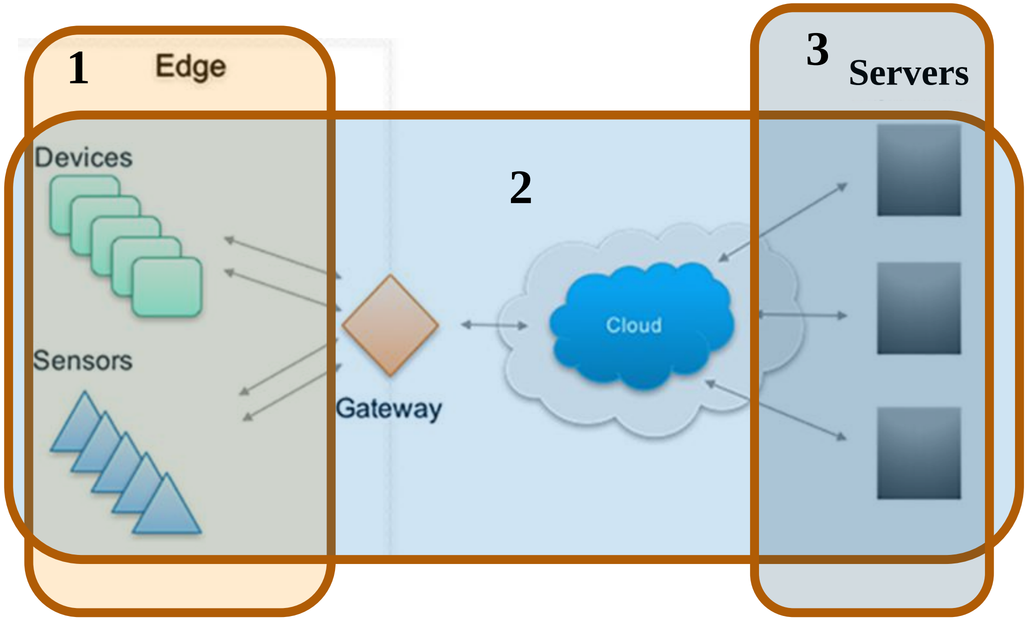
Perception Layer

Sensors and Actuators

Communication Requirements of IoT

- **Three** communication requirements to be addressed in IoT Systems:

1. Between devices on a local network.
2. Between devices through the Internet and between devices and servers.
3. Between servers that store data.



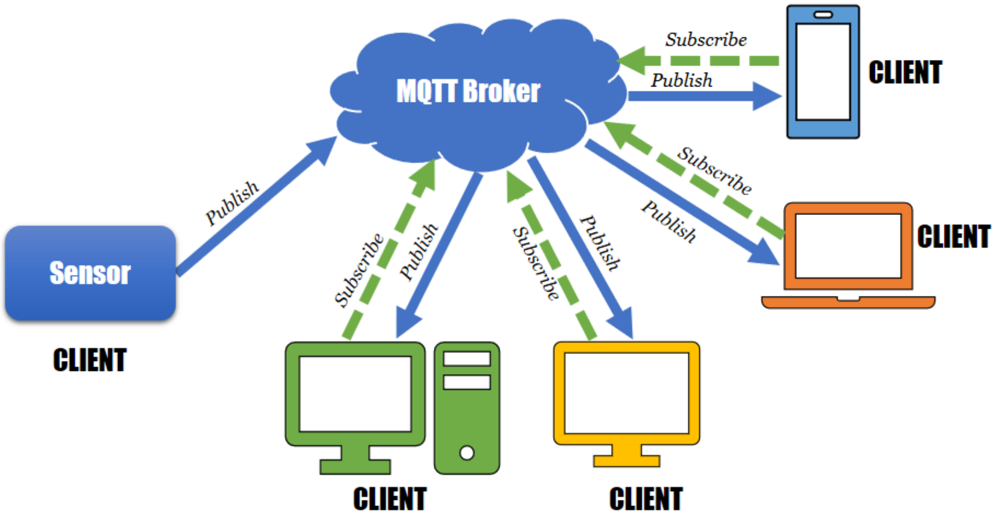
- There are a few **Application layer protocols** that solve the communication challenges faced by IoT systems which run on TCP or UDP.
- **MQTT** is one of them which runs on TCP.

MQTT:

Message Queue Telemetry Transport Protocol

Telemetry: It is the collection of measurements or other data at remote or inaccessible points and their automatic transmission for monitoring.

MQTT: Introduction



MQTT has **three entities**:
broker, subscribers and **publishers**.

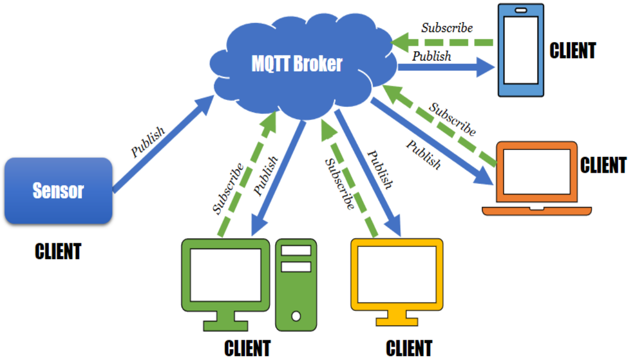
Publishers generate data based on **topics** and transmit them to **broker**, which shares them with **subscribers** to those topics**.**

The broker guarantees **safe** and **reliable** delivery of data from the **publishers** to all the **subscribers**.

- A publish/subscribe (pub/sub) communication model.
- A lightweight bi-directional open messaging protocol.
- Normally used for faraway tracking of IoT devices.
- Focus is to gather and deliver statistics from many gadgets.
- Resource-constrained network clients to distribute data in low-bandwidth environment.

MQTT: More Information

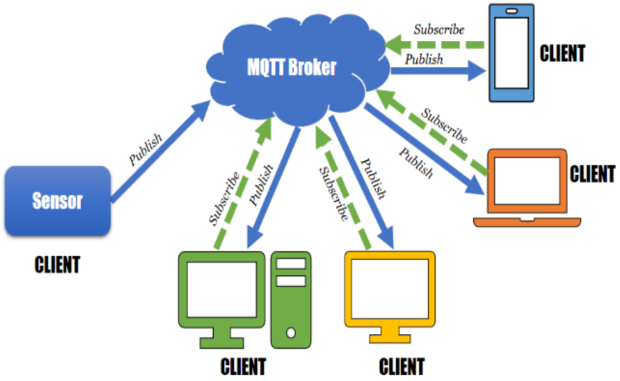
- MQTT was created in 1999 by two engineers - Andy Stanford-Clark (IBM) and Arlen Nipper (Eurotech).
- The protocol is an open OASIS standard and an ISO recommendation (ISO/IEC 20922).



- The latest version of the protocol specification is 5.0 as on 7 Mar 2019.
- **MQTT-SN** (MQTT for Sensor Networks) is a variation of the main protocol aimed at battery-powered embedded devices on non-TCP/IP networks, such as Zigbee, BLE, etc.
- There are no queues, in spite of the name, since it was specifically designed for resource-constrained devices and low bandwidth, high latency networks.
- The broker discards the message if there are no subscribers for that topic unless the publisher has marked it as **retained message**.
- The broker retains only one recent message per topic.

MQTT Protocol: TLS/SSL

- Plain MQTT sends connection credentials in plain text format.
- Security can be provided by using **TLS/SSL** to encrypt and protect the transferred information over Internet against interception, modification or forgery.



- Transport Layer Security (TLS) is an improved version of Secure Sockets Layer (SSL).
- The default unencrypted MQTT port is **1883**. The encrypted port is **8883**.
- **Username** and **password** are shared with MQTT CONNECT packet for authentication and authorization in TLS based MQTT.
- The broker shares certificate (CA file) in PEM (Privacy Enhanced Mail) format which is to be used by the clients to encrypt the messages before sending to MQTT broker.
- MQTT is scalable to 1000s of clients.
- Broker manages and tracks all client connection states, credentials and certificates.
- It has reduced network strain without compromising on security.

Mosquitto

An open source MQTT broker

mosquitto

Mosquitto: Installation Windows and Linux

- Eclipse Mosquitto is an open source (**EPL** licensed) message broker that implements the MQTT protocol versions 5.0, 3.1.1 and 3.1.
- Mosquitto is available on all platforms.
- https://mosquitto.org/download/
- Executable to install on Windows 10:
  - mosquitto-2.0.11-install-windows-x64.exe
- On Ubuntu or Raspberry Pi you can install Mosquitto broker with:
  - sudo apt-get update
  - sudo apt-get upgrade
  - sudo apt-get install mosquitto
- To view the command options: mosquitto –help
- To invoke mosquito broker in verbose mode: mosquitto –v
- To invoke the publisher: mosquitto_pub
- To invoke the subscriber: mosquitto_sub

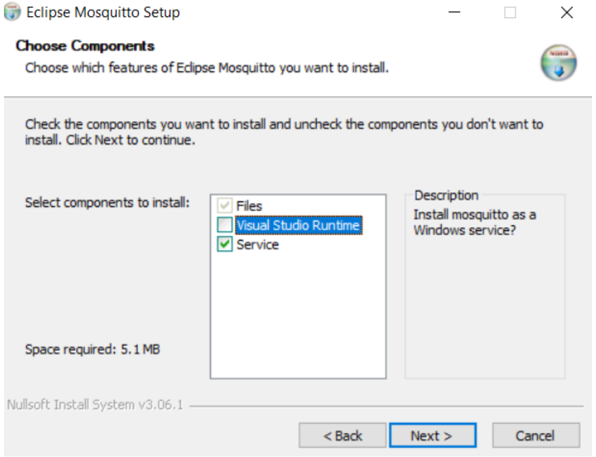
We will be
using the Windows
installation of Mosquitto
for our Demo

Mosquitto Manual

EPL: Eclipse Public License

Mosquitto: Installation on Windows … A Note

- While you run the installation executable (mosquitto-2.0.11-install-windows-x64.exe) on Windows, you will get an option to install Microsoft **Visual Studio Runtime** as well.
- To run Mosquitto on your machine, you don’t need to choose the above option.
- You are free to install Mosquitto without Visual Studio Runtime and you can still run Mosquitto without any issues.



Eclipse Mosquitto Setup

Choose Components

Choose which features of Eclipse Mosquitto you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

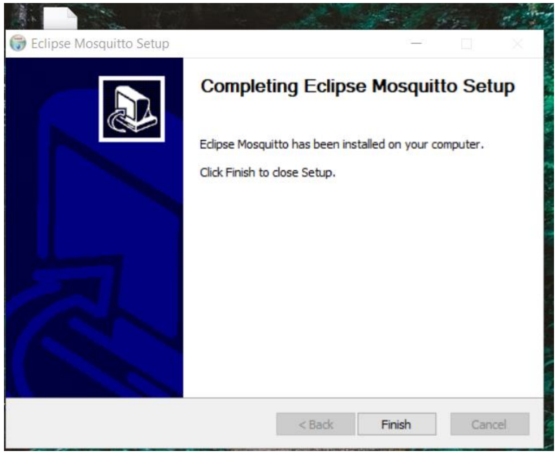
Select components to install:

☑ Files
□ Visual Studio Runtime
☑ Service

Description
Install mosquitto as a Windows service?

Space required: 5.1 MB

Nullsoft Install System v3.06.1



Eclipse Mosquitto Setup

Completing Eclipse Mosquitto Setup

Eclipse Mosquitto has been installed on your computer.

Click Finish to close Setup.

Mosquitto: Running the MQTT Broker on Windows

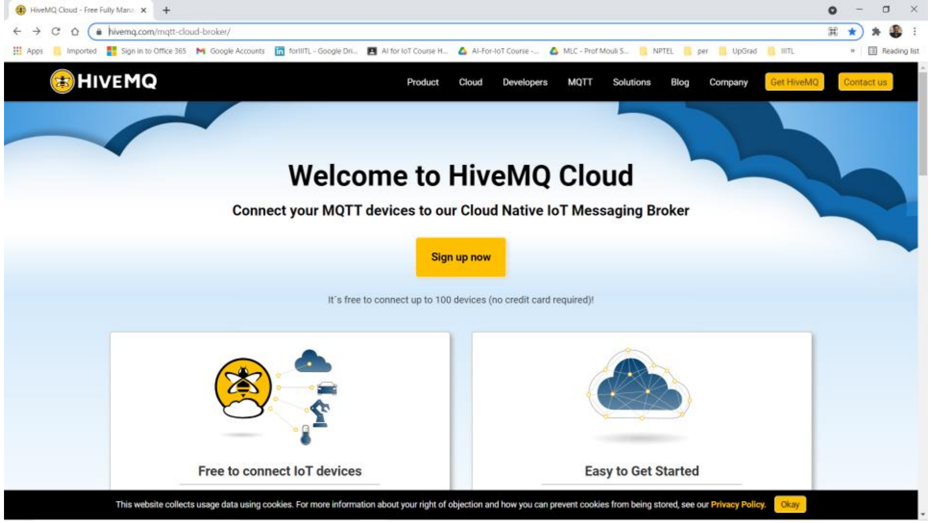
- After installing the Mosquitto on Windows, let us try running it.
- We can run the Broker on the local machine and communicate with it using subscriber and publisher of Mosquitto.
- Let us see a demo of it before we look at running it on the cloud using HiveMQ.
- Navigate to the directory where mosquitto is installed on your machine and give the following commands to invoke them on command prompt.
- To run the **broker** in verbose mode: mosquitto –v
- To run the **subscriber**: mosquitto_sub –h localhost –t Topic1
- To run the **publisher**: mosquitto_pub –h localhost – t Topic1 –m HiMsg

HiveMQ
An MQTT-based messaging platform

HIVEMQ

HiveMQ: MQTT Broker on Cloud: Login Steps

- Create **your account** with your **email** and a **password**. Login to HiveMQ
- Maximum two instances of MQTT brokers can be created for **free**.
- There are commercial and enterprise versions available with free trials.



Welcome to HiveMQ Cloud

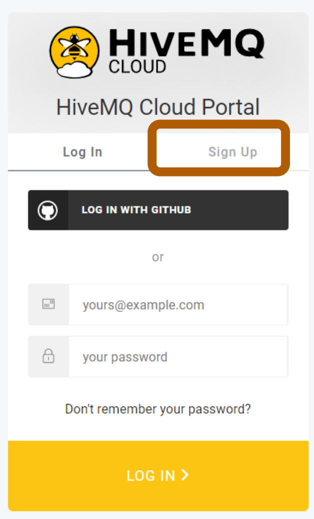
Connect your MQTT devices to our Cloud Native IoT Messaging Broker

It's free to connect up to 100 devices (no credit card required)!

Free to connect IoT devices

Easy to Get Started

This website collects usage data using cookies. For more information about your right of objection and how you can prevent cookies from being stored, see our Privacy Policy.

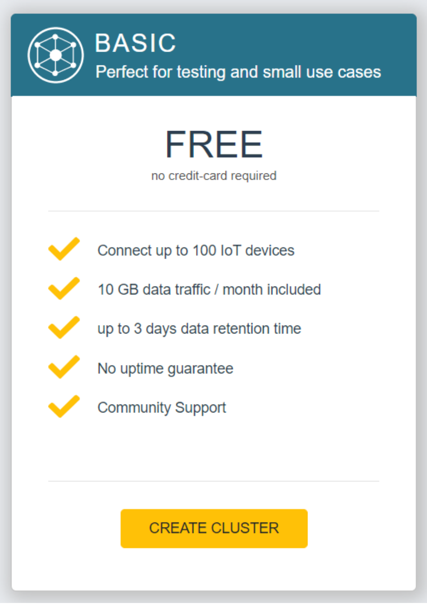


HiveMQ Cloud Portal

yours@example.com

your password

Don't remember your password?



BASIC

Perfect for testing and small use cases

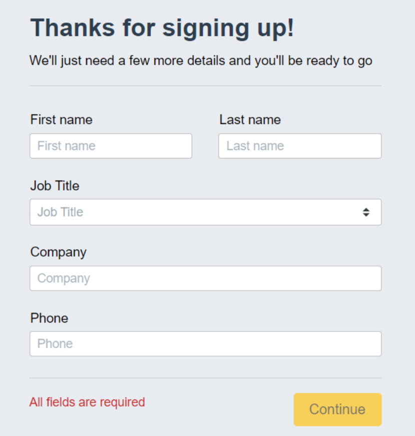
FREE

no credit-card required

- ✓ Connect up to 100 IoT devices
- ✓ 10 GB data traffic / month included
- ✓ up to 3 days data retention time
- ✓ No uptime guarantee
- ✓ Community Support

Note: After you sign up with your email id and a password, you need to confirm your email before HiveMQ allows you to login with the login credentials you have provided.

HiveMQ: MQTT Broker on Cloud: Registration



Thanks for signing up!

We'll just need a few more details and you'll be ready to go

First name: First name

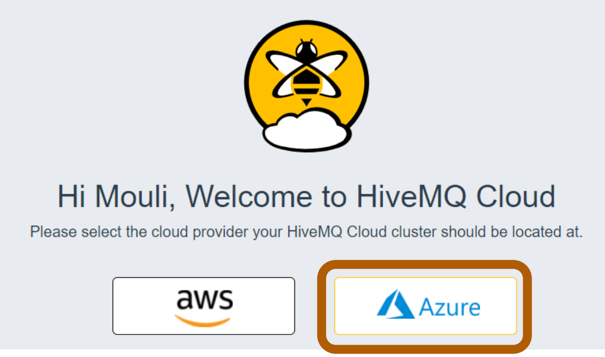
Last name: Last name

Job Title: Job Title

Company: Company

Phone: Phone

All fields are required

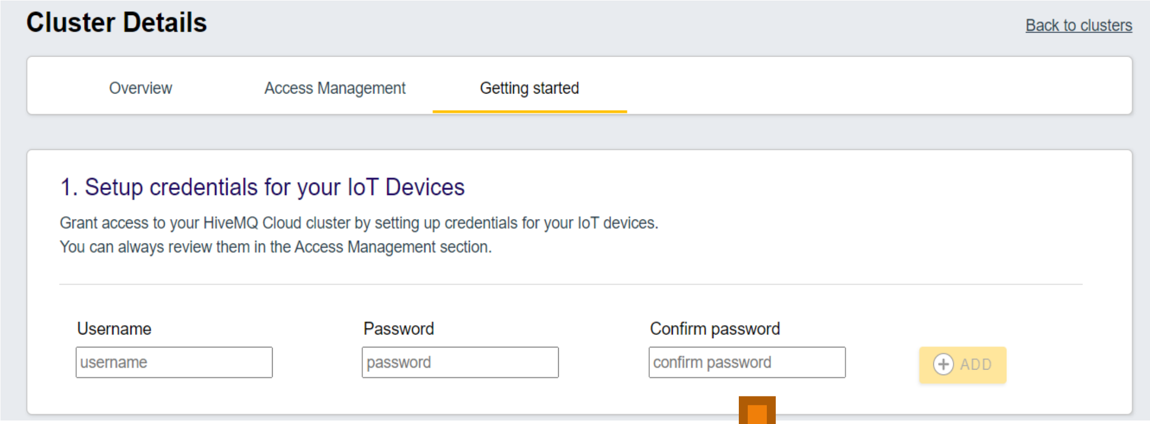


Hi Mouli, Welcome to HiveMQ Cloud

Please select the cloud provider your HiveMQ Cloud cluster should be located at.

You can create a MQTT broker on either AWS or Azure cloud.

You are free to choose any one of them based on your personal choice.



Cluster Details

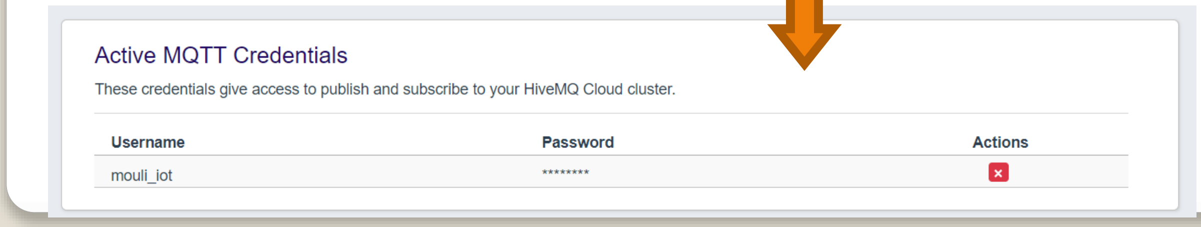
1. Setup credentials for your IoT Devices

Grant access to your HiveMQ Cloud cluster by setting up credentials for your IoT devices.
You can always review them in the Access Management section.

Username: username

Password: password

Confirm password: confirm password



Active MQTT Credentials

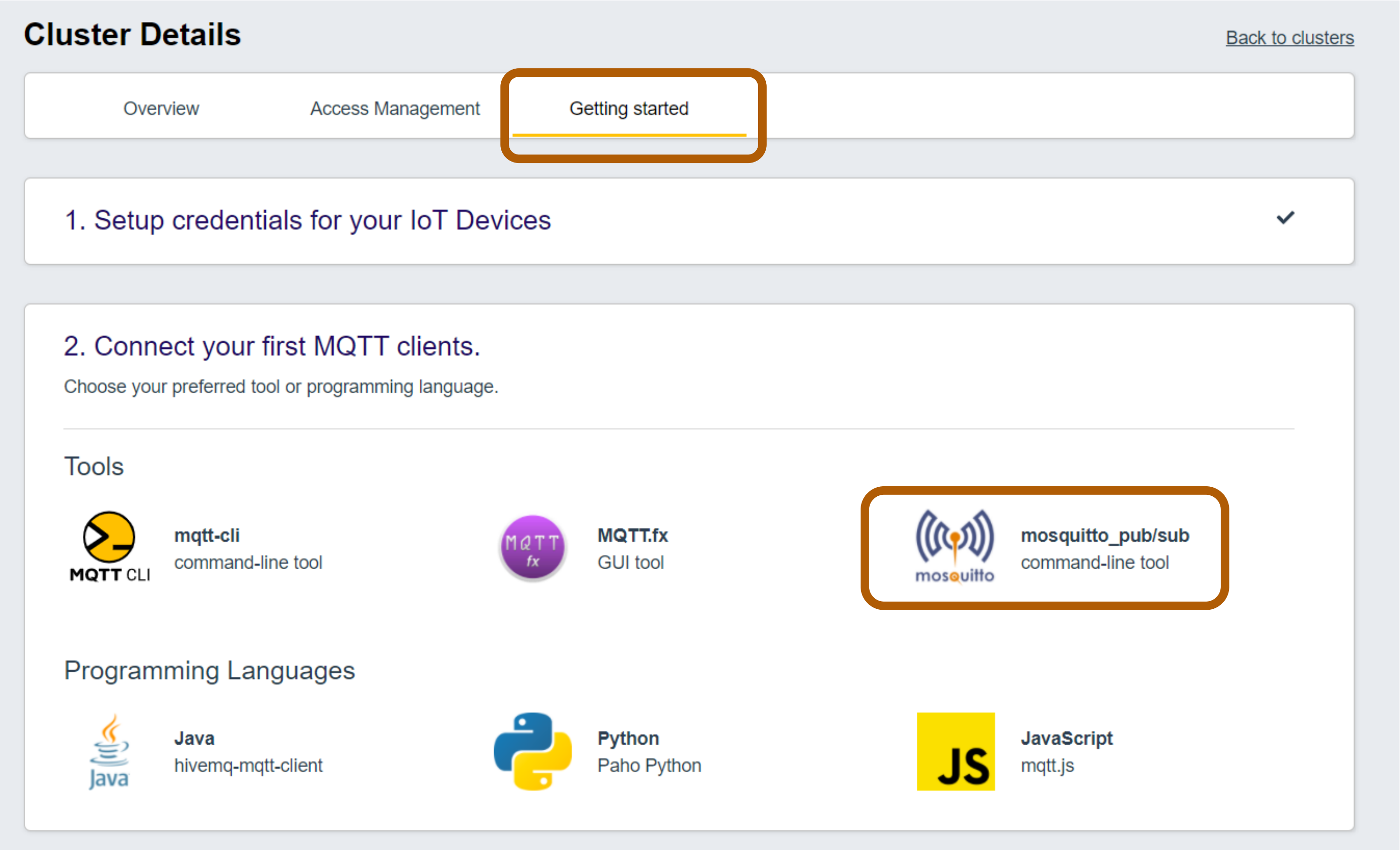
These credentials give access to publish and subscribe to your HiveMQ Cloud cluster.

Username	Password	Actions
mouli_iot	********	✕

Access Management

You have to provide your IoT device credentials: **username** and **password**.

HiveMQ: : MQTT Client Choices



Cluster Details

Back to clusters

1. Setup credentials for your IoT Devices ✓

2. Connect your first MQTT clients.

Choose your preferred tool or programming language.

Tools

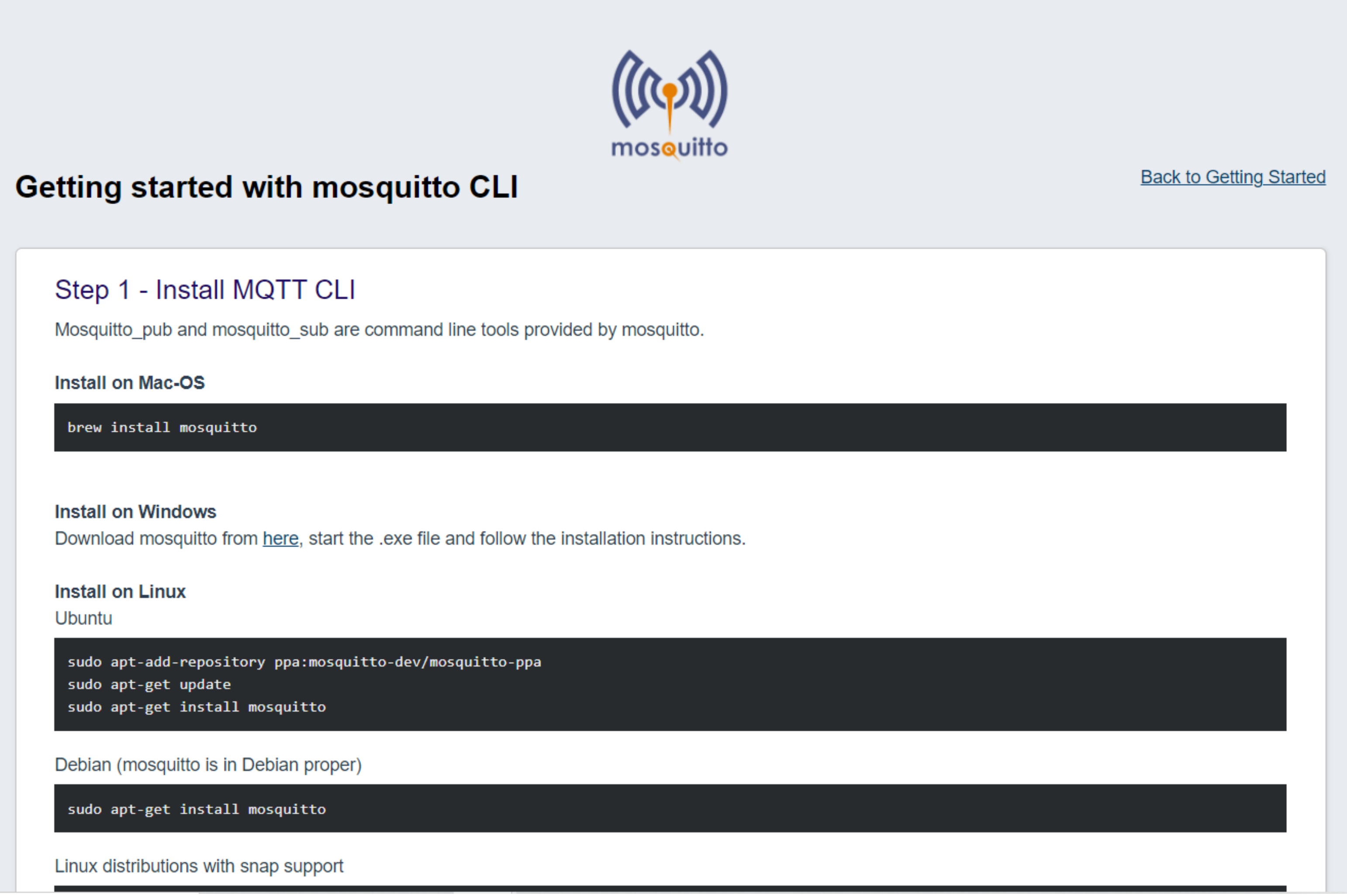
- **mqtt-cli** command-line tool
- **MQTT.fx** GUI tool
- **mosquitto_pub/sub** command-line tool

Programming Languages

- **Java** hivemq-mqtt-client
- **Python** Paho Python
- **JavaScript** mqtt.js

Note: We are choosing **Mosquitto**.

HiveMQ: : Getting Started with Mosquitto



mosquitto

Getting started with mosquitto CLI

Back to Getting Started

Step 1 - Install MQTT CLI

Mosquitto_pub and mosquitto_sub are command line tools provided by mosquitto.

Install on Mac-OS

```
brew install mosquitto
```

Install on Windows

Download mosquitto from here, start the .exe file and follow the installation instructions.

Install on Linux

Ubuntu

```
sudo apt-add-repository ppa:mosquitto-dev/mosquitto-ppa
sudo apt-get update
sudo apt-get install mosquitto
```

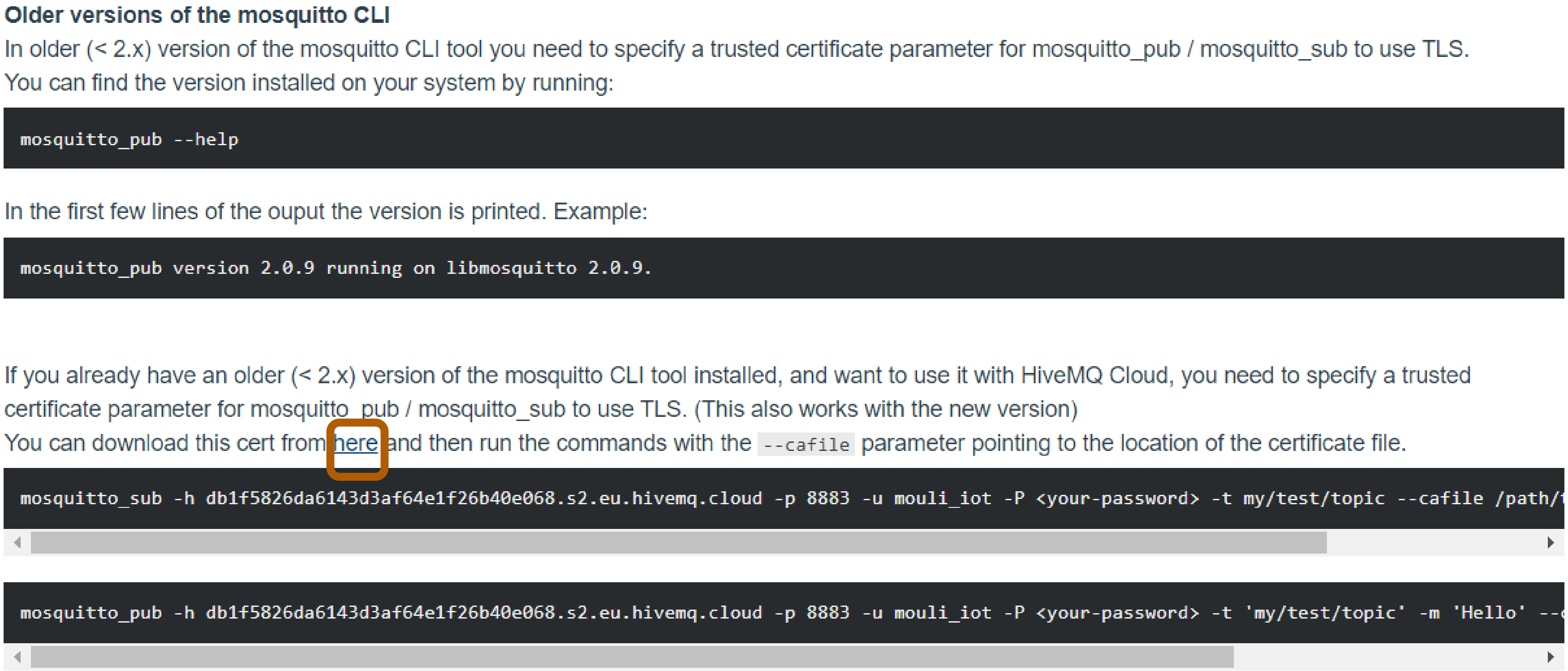
Debian (mosquitto is in Debian proper)

```
sudo apt-get install mosquitto
```

Linux distributions with snap support

Note: Scroll down to download the Certificate.

HiveMQ: : Download Certificate (in .pem format)



Older versions of the mosquitto CLI

In older (< 2.x) version of the mosquitto CLI tool you need to specify a trusted certificate parameter for mosquitto_pub / mosquitto_sub to use TLS.
You can find the version installed on your system by running:

```
mosquitto_pub --help
```

In the first few lines of the ouput the version is printed. Example:

```
mosquitto_pub version 2.0.9 running on libmosquitto 2.0.9.
```

If you already have an older (< 2.x) version of the mosquitto CLI tool installed, and want to use it with HiveMQ Cloud, you need to specify a trusted certificate parameter for mosquitto_pub / mosquitto_sub to use TLS. (This also works with the new version)
You can download this cert from here and then run the commands with the `--cafile` parameter pointing to the location of the certificate file.

```
mosquitto_sub -h db1f5826da6143d3af64e1f26b40e068.s2.eu.hivemq.cloud -p 8883 -u mouli_iot -P <your-password> -t my/test/topic --cafile /path/t
```

```
mosquitto_pub -h db1f5826da6143d3af64e1f26b40e068.s2.eu.hivemq.cloud -p 8883 -u mouli_iot -P <your-password> -t 'my/test/topic' -m 'Hello' --c
```

Note 1: Click here and download the .pem file and store it in your machine.

Note 2: You can learn about how to communicate with HiveMQ using Mosquitto clients, here.

Summary: IoT - Tutorial 1

IoT – Tutorial 0 – ESP32 Installation Guide

- MQTT Protocol
  - Introduction
  - Features
- Mosquitto
  - Installation on Windows 10
  - Usage
- MQTT on Cloud - HiveMQ
  - HiveMQ Registration
  - Getting started with HiveMQ
  - Demo

IoT – Tutorial 2 – Porting MQTT Client on to ESP32 Board