

Project Work IoT



19/7/2023

Matteo Andruccioli

Obiettivi del progetto

Obiettivo principale: Progettare e configurare un'infrastruttura Cloud utilizzando Amazon Web Services (AWS) per un'azienda produttrice di impianti di deumidificazione.

Scopo del progetto: Consentire il monitoraggio in tempo reale dei dati provenienti dalle schede dei dispositivi dell'azienda e consentire la gestione remota dell'impianto.

Dettagli dell'obiettivo:

- Progettare un sistema che permetta alle schede dei dispositivi di connettersi a un gateway per la raccolta dei dati provenienti dai sensori.
- Invio dei dati raccolti al cloud AWS per l'elaborazione e la visualizzazione.
- Implementazione di un'architettura scalabile, sicura e resiliente utilizzando i servizi di AWS.
- Configurazione di strumenti di monitoraggio e gestione per garantire il corretto funzionamento dell'infrastruttura.
- Creazione di un'interfaccia utente intuitiva per la visualizzazione dei dati e l'invio di comandi remoti per la gestione dell'impianto.

Benefici attesi:

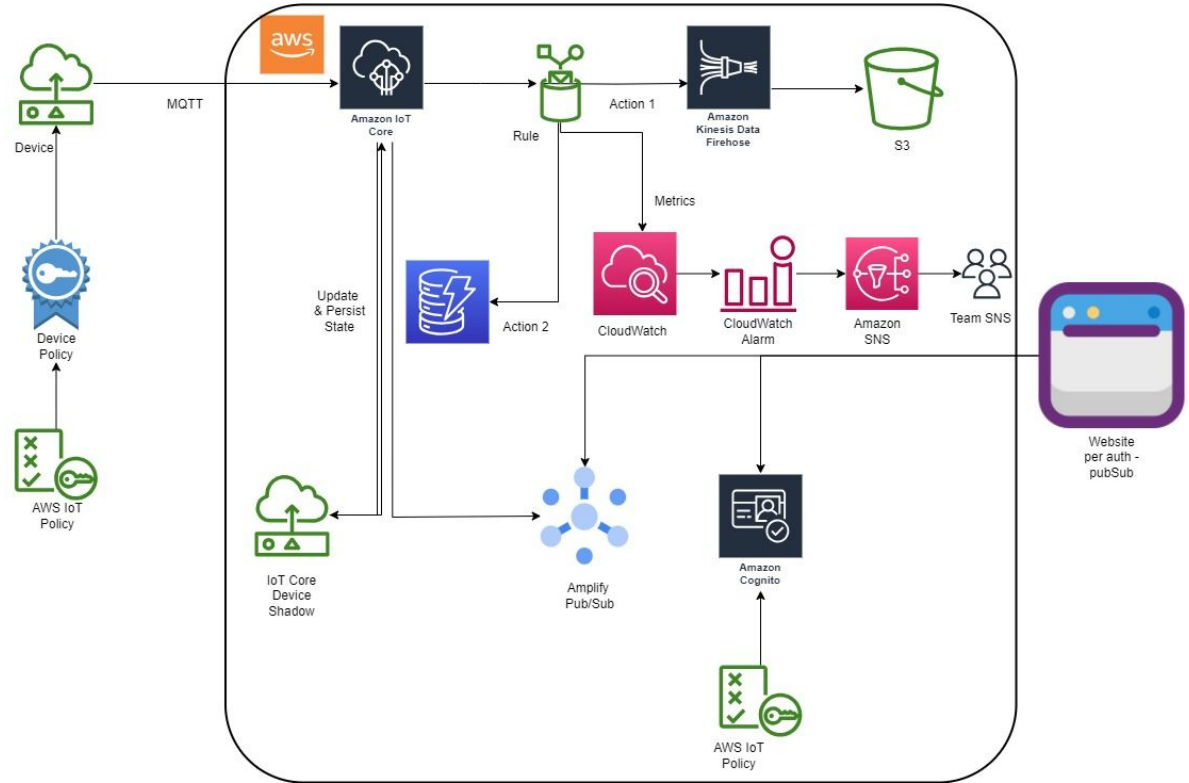
- Monitoraggio in tempo reale dei dati dei sensori per un'analisi immediata delle prestazioni dell'impianto.
- Possibilità di gestire l'impianto in remoto, consentendo interventi rapidi e migliorando l'efficienza operativa.
- Archiviazione dei dati su cloud per una maggiore scalabilità, flessibilità e disponibilità.
- Utilizzo di servizi e risorse su AWS per ridurre i costi di gestione e manutenzione dell'infrastruttura.

Output attesi:

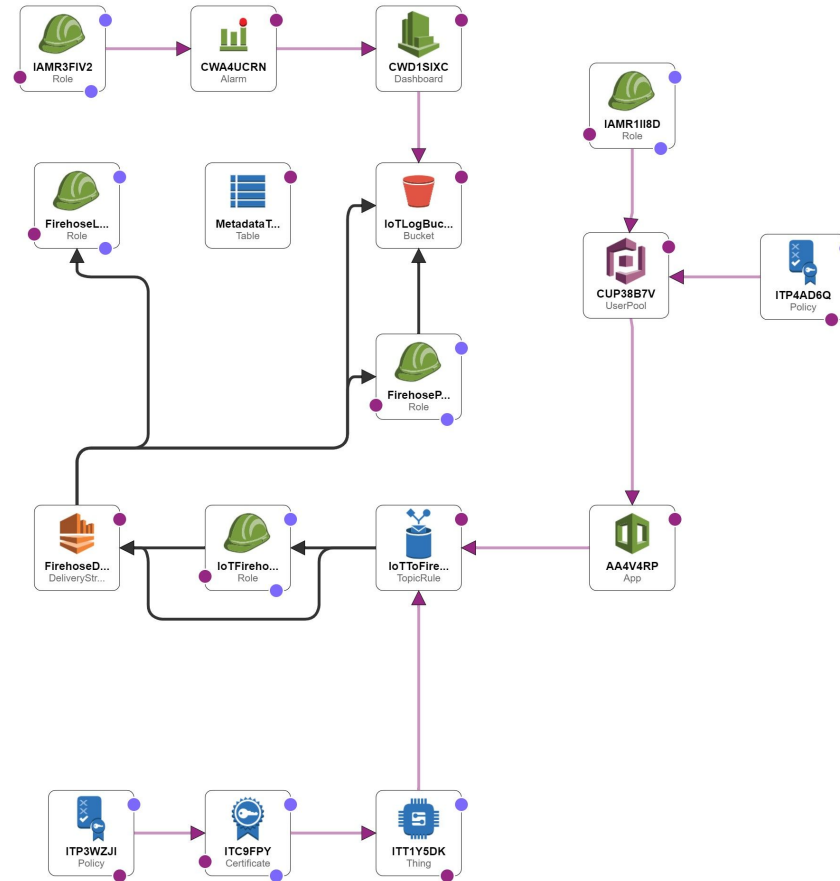
- Documento di progetto con assunzioni fare per la realizzazione della proposta fatta e calculator dei costi teorici di infrastruttura.
- Template CDK dell'architettura.
- Codice sorgente degli script scritti per simulare un impianto.
- Screenshot cost explorer dei costi di infrastruttura reali e confronto con il calcolo teorico mappando eventuali azioni di miglioramento.

Architettura generale

- IoT Core
- Kinesis Firehose
- Amazon S3
- DynamoDB
- Amazon Cognito
- Amplify
- Cloudwatch
- SNS
- IAM



Architettura Cloudformation



Raccolta dei dati

Per la raccolta dei dati ho utilizzato AWS IoT Core, un servizio di gestione e connettività per dispositivi IoT che offre una soluzione scalabile e sicura per la raccolta dei dati dai dispositivi e la loro trasmissione al cloud AWS. Nel contesto del progetto, AWS IoT Core svolge un ruolo fondamentale nel consentire la connessione dei dispositivi al cloud AWS per la raccolta dei dati in tempo reale.

I vantaggi principali nell'utilizzare AWS IoT Core includono:

1. **Connettività sicura:** AWS IoT Core offre funzionalità di sicurezza avanzate per proteggere la connessione tra i dispositivi IoT e il cloud AWS. Supporta la crittografia end-to-end dei dati, l'autenticazione dei dispositivi e il controllo degli accessi, garantendo che solo i dispositivi autorizzati possano inviare dati al cloud.
2. **Scalabilità:** AWS IoT Core consente di gestire facilmente un gran numero di dispositivi IoT. Può scalare automaticamente per gestire picchi di connessioni e volume dei dati. Questo ci permette di espandere il nostro progetto senza dover preoccuparci della gestione dei dispositivi.
3. **Gestione semplificata dei dispositivi:** AWS IoT Core fornisce strumenti per la registrazione, l'autenticazione e la configurazione dei dispositivi IoT. E' facilmente possibile registrare i dispositivi nel servizio e gestire in modo centralizzato le loro autorizzazioni e configurazioni. Ciò semplifica la gestione e il monitoraggio dei dispositivi in modo efficiente.
4. **Protocolli di comunicazione flessibili**
5. **Regole di routing**

Simulazione ed invio dati

Simulazione ed invio dati tramite MQTT utilizzando la libreria AWSIoTPythonSDK da una Ec2 ad Aws IoT.

L'invio può essere visualizzato nella sezione Subscriptions, per farlo è stato necessario configurare l'endpoint del servizio ed importare i certificati del dispositivo che stiamo simulando.

The image displays the AWS IoT console interface on the left and a terminal window on the right, illustrating the setup and execution of an MQTT simulation.

AWS IoT Console (Left):

- Topic filter:** Set to `general/device1/typea`. The filter can include MQTT wildcard characters.
- Subscriptions:** Shows two active subscriptions for `general/device1/typea`, both created on July 17, 2023, at 09:24:30 (UTC+0200).
- Subscription 1:** The message payload is a JSON object:

```
{  "SeqNumber": 99,  "message": "Test Message - 100",  "val1": "Value 3 - 100",  "val2": "Value 2 - 100",  "val3": "Value 1 - 100",  "SeqSort": 1}
```
- Subscription 2:** The message payload is a JSON object:

```
{  "SeqNumber": 98,  "message": "Test Message - 99",  "val1": "Value 3 - 99",  "val2": "Value 2 - 99",  "val3": "Value 1 - 99",  "SeqSort": 1}
```

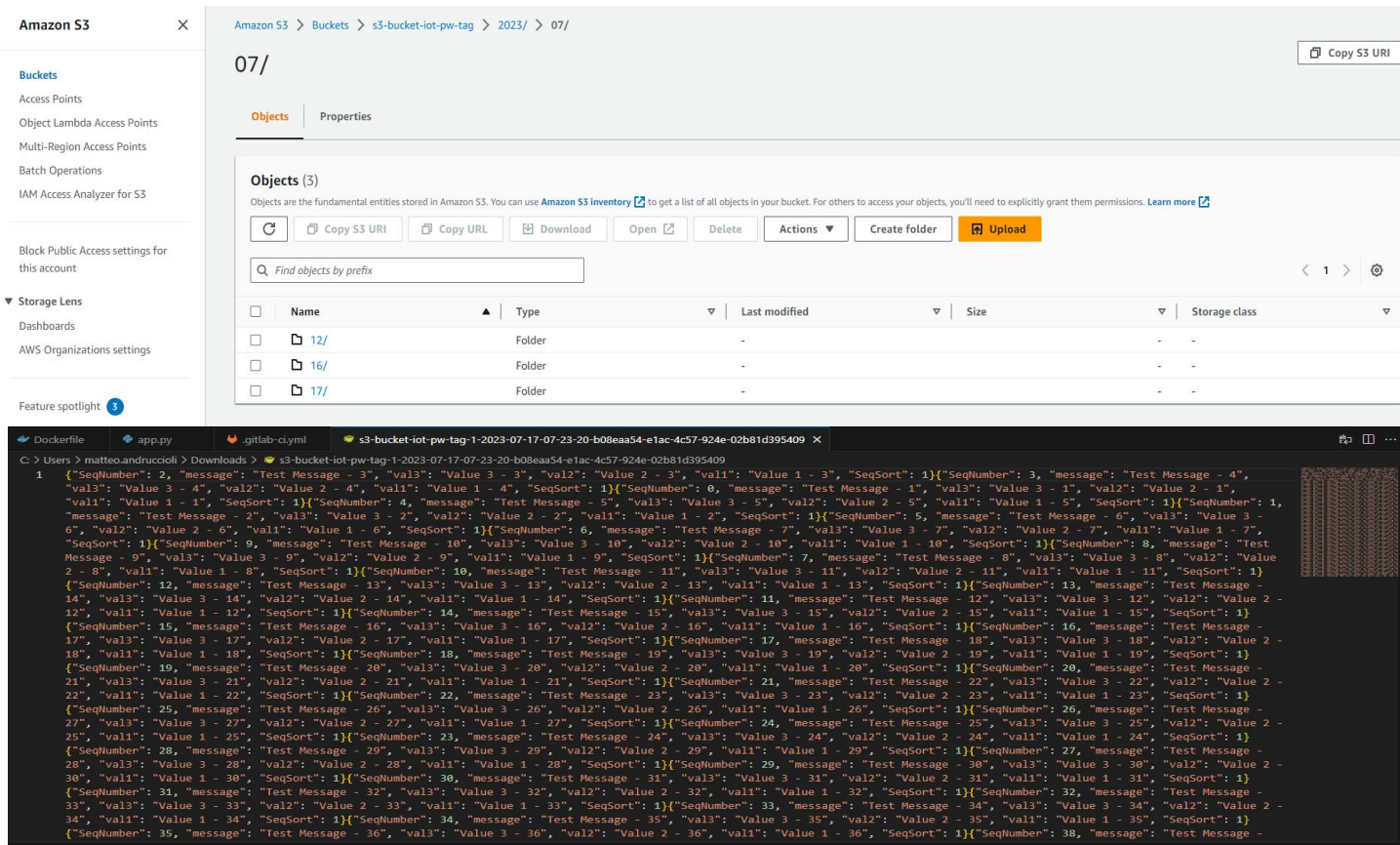
Terminal Window (Right):

- File Explorer:** Shows the directory structure for the simulation, including `aws-iot-device-sdk-python`, `05583444acde1272258804ee1e0a550cdeb4`, `AmazonRootCA1.pem`, `AmazonRootCA3.pem`, `publish.py`, `publish2.py`, `publishOld.py`, `README.md`, and `subscribe.py`.
- Code Editor:** Displays the `publish.py` script, which uses the `AWSIoTPythonSDK` to connect to the AWS IoT endpoint and publish messages to the `general/device1/typea` topic. The script includes comments for setting up the MQTT client and security certificates, and a loop to generate and publish data.
- Terminal Output:** Shows the execution of the `publish.py` script, displaying the published messages and their payloads, matching the data shown in the AWS IoT console.

Archiviazione dati (1/2)

L'archiviazione dei dati è stato un aspetto fondamentale per garantire la persistenza e l'accessibilità dei dati raccolti dai dispositivi. Nel contesto del progetto, utilizziamo **Amazon S3** e **Amazon DynamoDB** come soluzioni per l'archiviazione dei dati raccolti.

Download ed apertura Object S3 con i dati ricevuti in Mqtt



The screenshot displays the Amazon S3 console interface. The top navigation bar shows the path: Amazon S3 > Buckets > s3-bucket-iot-pw-tag > 2023/ > 07/. The left sidebar contains the navigation menu with options like Buckets, Access Points, Object Lambda Access Points, Multi-Region Access Points, Batch Operations, IAM Access Analyzer for S3, Storage Lens, Dashboards, AWS Organizations settings, and Feature spotlight. The main content area shows the 'Objects (3)' section, indicating that there are three objects in the bucket. Below this, there are buttons for 'Copy S3 URI', 'Copy URL', 'Download', 'Open', 'Delete', 'Actions', 'Create folder', and 'Upload'. A search bar is present with the placeholder text 'Find objects by prefix'. The table below lists the objects:

	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	12/	Folder	-	-	-
<input type="checkbox"/>	16/	Folder	-	-	-
<input type="checkbox"/>	17/	Folder	-	-	-

Below the console, a terminal window shows the contents of the 12/ folder, which are 35 JSON objects representing test messages. The first object is:

```
1 {
  "SeqNumber": 2, "message": "Test Message - 3", "val3": "Value 3 - 3", "val2": "Value 2 - 3", "val1": "Value 1 - 3", "SeqSort": 1}{"SeqNumber": 3, "message": "Test Message - 4",
  "val3": "Value 3 - 4", "val2": "Value 2 - 4", "val1": "Value 1 - 4", "SeqSort": 1}{"SeqNumber": 4, "message": "Test Message - 5", "val3": "Value 3 - 5", "val2": "Value 2 - 5", "val1": "Value 1 - 5", "SeqSort": 1}{"SeqNumber": 5, "message": "Test Message - 6", "val3": "Value 3 - 6", "val2": "Value 2 - 6", "val1": "Value 1 - 6", "SeqSort": 1}{"SeqNumber": 6, "message": "Test Message - 7", "val3": "Value 3 - 7", "val2": "Value 2 - 7", "val1": "Value 1 - 7", "SeqSort": 1}{"SeqNumber": 7, "message": "Test Message - 8", "val3": "Value 3 - 8", "val2": "Value 2 - 8", "val1": "Value 1 - 8", "SeqSort": 1}{"SeqNumber": 8, "message": "Test Message - 9", "val3": "Value 3 - 9", "val2": "Value 2 - 9", "val1": "Value 1 - 9", "SeqSort": 1}{"SeqNumber": 9, "message": "Test Message - 10", "val3": "Value 3 - 10", "val2": "Value 2 - 10", "val1": "Value 1 - 10", "SeqSort": 1}{"SeqNumber": 10, "message": "Test Message - 11", "val3": "Value 3 - 11", "val2": "Value 2 - 11", "val1": "Value 1 - 11", "SeqSort": 1}{"SeqNumber": 11, "message": "Test Message - 12", "val3": "Value 3 - 12", "val2": "Value 2 - 12", "val1": "Value 1 - 12", "SeqSort": 1}{"SeqNumber": 12, "message": "Test Message - 13", "val3": "Value 3 - 13", "val2": "Value 2 - 13", "val1": "Value 1 - 13", "SeqSort": 1}{"SeqNumber": 13, "message": "Test Message - 14", "val3": "Value 3 - 14", "val2": "Value 2 - 14", "val1": "Value 1 - 14", "SeqSort": 1}{"SeqNumber": 14, "message": "Test Message - 15", "val3": "Value 3 - 15", "val2": "Value 2 - 15", "val1": "Value 1 - 15", "SeqSort": 1}{"SeqNumber": 15, "message": "Test Message - 16", "val3": "Value 3 - 16", "val2": "Value 2 - 16", "val1": "Value 1 - 16", "SeqSort": 1}{"SeqNumber": 16, "message": "Test Message - 17", "val3": "Value 3 - 17", "val2": "Value 2 - 17", "val1": "Value 1 - 17", "SeqSort": 1}{"SeqNumber": 17, "message": "Test Message - 18", "val3": "Value 3 - 18", "val2": "Value 2 - 18", "val1": "Value 1 - 18", "SeqSort": 1}{"SeqNumber": 18, "message": "Test Message - 19", "val3": "Value 3 - 19", "val2": "Value 2 - 19", "val1": "Value 1 - 19", "SeqSort": 1}{"SeqNumber": 19, "message": "Test Message - 20", "val3": "Value 3 - 20", "val2": "Value 2 - 20", "val1": "Value 1 - 20", "SeqSort": 1}{"SeqNumber": 20, "message": "Test Message - 21", "val3": "Value 3 - 21", "val2": "Value 2 - 21", "val1": "Value 1 - 21", "SeqSort": 1}{"SeqNumber": 21, "message": "Test Message - 22", "val3": "Value 3 - 22", "val2": "Value 2 - 22", "val1": "Value 1 - 22", "SeqSort": 1}{"SeqNumber": 22, "message": "Test Message - 23", "val3": "Value 3 - 23", "val2": "Value 2 - 23", "val1": "Value 1 - 23", "SeqSort": 1}{"SeqNumber": 23, "message": "Test Message - 24", "val3": "Value 3 - 24", "val2": "Value 2 - 24", "val1": "Value 1 - 24", "SeqSort": 1}{"SeqNumber": 24, "message": "Test Message - 25", "val3": "Value 3 - 25", "val2": "Value 2 - 25", "val1": "Value 1 - 25", "SeqSort": 1}{"SeqNumber": 25, "message": "Test Message - 26", "val3": "Value 3 - 26", "val2": "Value 2 - 26", "val1": "Value 1 - 26", "SeqSort": 1}{"SeqNumber": 26, "message": "Test Message - 27", "val3": "Value 3 - 27", "val2": "Value 2 - 27", "val1": "Value 1 - 27", "SeqSort": 1}{"SeqNumber": 27, "message": "Test Message - 28", "val3": "Value 3 - 28", "val2": "Value 2 - 28", "val1": "Value 1 - 28", "SeqSort": 1}{"SeqNumber": 28, "message": "Test Message - 29", "val3": "Value 3 - 29", "val2": "Value 2 - 29", "val1": "Value 1 - 29", "SeqSort": 1}{"SeqNumber": 29, "message": "Test Message - 30", "val3": "Value 3 - 30", "val2": "Value 2 - 30", "val1": "Value 1 - 30", "SeqSort": 1}{"SeqNumber": 30, "message": "Test Message - 31", "val3": "Value 3 - 31", "val2": "Value 2 - 31", "val1": "Value 1 - 31", "SeqSort": 1}{"SeqNumber": 31, "message": "Test Message - 32", "val3": "Value 3 - 32", "val2": "Value 2 - 32", "val1": "Value 1 - 32", "SeqSort": 1}{"SeqNumber": 32, "message": "Test Message - 33", "val3": "Value 3 - 33", "val2": "Value 2 - 33", "val1": "Value 1 - 33", "SeqSort": 1}{"SeqNumber": 33, "message": "Test Message - 34", "val3": "Value 3 - 34", "val2": "Value 2 - 34", "val1": "Value 1 - 34", "SeqSort": 1}{"SeqNumber": 34, "message": "Test Message - 35", "val3": "Value 3 - 35", "val2": "Value 2 - 35", "val1": "Value 1 - 35", "SeqSort": 1}{"SeqNumber": 35, "message": "Test Message - 36", "val3": "Value 3 - 36", "val2": "Value 2 - 36", "val1": "Value 1 - 36", "SeqSort": 1}{"SeqNumber": 36, "message": "Test Message - 37", "val3": "Value 3 - 37", "val2": "Value 2 - 37", "val1": "Value 1 - 37", "SeqSort": 1}{"SeqNumber": 37, "message": "Test Message - 38", "val3": "Value 3 - 38", "val2": "Value 2 - 38", "val1": "Value 1 - 38", "SeqSort": 1}{"SeqNumber": 38, "message": "Test Message - 39", "val3": "Value 3 - 39", "val2": "Value 2 - 39", "val1": "Value 1 - 39", "SeqSort": 1}{"SeqNumber": 39, "message": "Test Message - 40", "val3": "Value 3 - 40", "val2": "Value 2 - 40", "val1": "Value 1 - 40", "SeqSort": 1}{"SeqNumber": 40, "message": "Test Message - 41", "val3": "Value 3 - 41", "val2": "Value 2 - 41", "val1": "Value 1 - 41", "SeqSort": 1}{"SeqNumber": 41, "message": "Test Message - 42", "val3": "Value 3 - 42", "val2": "Value 2 - 42", "val1": "Value 1 - 42", "SeqSort": 1}{"SeqNumber": 42, "message": "Test Message - 43", "val3": "Value 3 - 43", "val2": "Value 2 - 43", "val1": "Value 1 - 43", "SeqSort": 1}{"SeqNumber": 43, "message": "Test Message - 44", "val3": "Value 3 - 44", "val2": "Value 2 - 44", "val1": "Value 1 - 44", "SeqSort": 1}{"SeqNumber": 44, "message": "Test Message - 45", "val3": "Value 3 - 45", "val2": "Value 2 - 45", "val1": "Value 1 - 45", "SeqSort": 1}{"SeqNumber": 45, "message": "Test Message - 46", "val3": "Value 3 - 46", "val2": "Value 2 - 46", "val1": "Value 1 - 46", "SeqSort": 1}{"SeqNumber": 46, "message": "Test Message - 47", "val3": "Value 3 - 47", "val2": "Value 2 - 47", "val1": "Value 1 - 47", "SeqSort": 1}{"SeqNumber": 47, "message": "Test Message - 48", "val3": "Value 3 - 48", "val2": "Value 2 - 48", "val1": "Value 1 - 48", "SeqSort": 1}{"SeqNumber": 48, "message": "Test Message - 49", "val3": "Value 3 - 49", "val2": "Value 2 - 49", "val1": "Value 1 - 49", "SeqSort": 1}{"SeqNumber": 49, "message": "Test Message - 50", "val3": "Value 3 - 50", "val2": "Value 2 - 50", "val1": "Value 1 - 50", "SeqSort": 1}{"SeqNumber": 50, "message": "Test Message - 51", "val3": "Value 3 - 51", "val2": "Value 2 - 51", "val1": "Value 1 - 51", "SeqSort": 1}{"SeqNumber": 51, "message": "Test Message - 52", "val3": "Value 3 - 52", "val2": "Value 2 - 52", "val1": "Value 1 - 52", "SeqSort": 1}{"SeqNumber": 52, "message": "Test Message - 53", "val3": "Value 3 - 53", "val2": "Value 2 - 53", "val1": "Value 1 - 53", "SeqSort": 1}{"SeqNumber": 53, "message": "Test Message - 54", "val3": "Value 3 - 54", "val2": "Value 2 - 54", "val1": "Value 1 - 54", "SeqSort": 1}{"SeqNumber": 54, "message": "Test Message - 55", "val3": "Value 3 - 55", "val2": "Value 2 - 55", "val1": "Value 1 - 55", "SeqSort": 1}{"SeqNumber": 55, "message": "Test Message - 56", "val3": "Value 3 - 56", "val2": "Value 2 - 56", "val1": "Value 1 - 56", "SeqSort": 1}{"SeqNumber": 56, "message": "Test Message - 57", "val3": "Value 3 - 57", "val2": "Value 2 - 57", "val1": "Value 1 - 57", "SeqSort": 1}{"SeqNumber": 57, "message": "Test Message - 58", "val3": "Value 3 - 58", "val2": "Value 2 - 58", "val1": "Value 1 - 58", "SeqSort": 1}{"SeqNumber": 58, "message": "Test Message - 59", "val3": "Value 3 - 59", "val2": "Value 2 - 59", "val1": "Value 1 - 59", "SeqSort": 1}{"SeqNumber": 59, "message": "Test Message - 60", "val3": "Value 3 - 60", "val2": "Value 2 - 60", "val1": "Value 1 - 60", "SeqSort": 1}{"SeqNumber": 60, "message": "Test Message - 61", "val3": "Value 3 - 61", "val2": "Value 2 - 61", "val1": "Value 1 - 61", "SeqSort": 1}{"SeqNumber": 61, "message": "Test Message - 62", "val3": "Value 3 - 62", "val2": "Value 2 - 62", "val1": "Value 1 - 62", "SeqSort": 1}{"SeqNumber": 62, "message": "Test Message - 63", "val3": "Value 3 - 63", "val2": "Value 2 - 63", "val1": "Value 1 - 63", "SeqSort": 1}{"SeqNumber": 63, "message": "Test Message - 64", "val3": "Value 3 - 64", "val2": "Value 2 - 64", "val1": "Value 1 - 64", "SeqSort": 1}{"SeqNumber": 64, "message": "Test Message - 65", "val3": "Value 3 - 65", "val2": "Value 2 - 65", "val1": "Value 1 - 65", "SeqSort": 1}{"SeqNumber": 65, "message": "Test Message - 66", "val3": "Value 3 - 66", "val2": "Value 2 - 66", "val1": "Value 1 - 66", "SeqSort": 1}{"SeqNumber": 66, "message": "Test Message - 67", "val3": "Value 3 - 67", "val2": "Value 2 - 67", "val1": "Value 1 - 67", "SeqSort": 1}{"SeqNumber": 67, "message": "Test Message - 68", "val3": "Value 3 - 68", "val2": "Value 2 - 68", "val1": "Value 1 - 68", "SeqSort": 1}{"SeqNumber": 68, "message": "Test Message - 69", "val3": "Value 3 - 69", "val2": "Value 2 - 69", "val1": "Value 1 - 69", "SeqSort": 1}{"SeqNumber": 69, "message": "Test Message - 70", "val3": "Value 3 - 70", "val2": "Value 2 - 70", "val1": "Value 1 - 70", "SeqSort": 1}{"SeqNumber": 70, "message": "Test Message - 71", "val3": "Value 3 - 71", "val2": "Value 2 - 71", "val1": "Value 1 - 71", "SeqSort": 1}{"SeqNumber": 71, "message": "Test Message - 72", "val3": "Value 3 - 72", "val2": "Value 2 - 72", "val1": "Value 1 - 72", "SeqSort": 1}{"SeqNumber": 72, "message": "Test Message - 73", "val3": "Value 3 - 73", "val2": "Value 2 - 73", "val1": "Value 1 - 73", "SeqSort": 1}{"SeqNumber": 73, "message": "Test Message - 74", "val3": "Value 3 - 74", "val2": "Value 2 - 74", "val1": "Value 1 - 74", "SeqSort": 1}{"SeqNumber": 74, "message": "Test Message - 75", "val3": "Value 3 - 75", "val2": "Value 2 - 75", "val1": "Value 1 - 75", "SeqSort": 1}{"SeqNumber": 75, "message": "Test Message - 76", "val3": "Value 3 - 76", "val2": "Value 2 - 76", "val1": "Value 1 - 76", "SeqSort": 1}{"SeqNumber": 76, "message": "Test Message - 77", "val3": "Value 3 - 77", "val2": "Value 2 - 77", "val1": "Value 1 - 77", "SeqSort": 1}{"SeqNumber": 77, "message": "Test Message - 78", "val3": "Value 3 - 78", "val2": "Value 2 - 78", "val1": "Value 1 - 78", "SeqSort": 1}{"SeqNumber": 78, "message": "Test Message - 79", "val3": "Value 3 - 79", "val2": "Value 2 - 79", "val1": "Value 1 - 79", "SeqSort": 1}{"SeqNumber": 79, "message": "Test Message - 80", "val3": "Value 3 - 80", "val2": "Value 2 - 80", "val1": "Value 1 - 80", "SeqSort": 1}{"SeqNumber": 80, "message": "Test Message - 81", "val3": "Value 3 - 81", "val2": "Value 2 - 81", "val1": "Value 1 - 81", "SeqSort": 1}{"SeqNumber": 81, "message": "Test Message - 82", "val3": "Value 3 - 82", "val2": "Value 2 - 82", "val1": "Value 1 - 82", "SeqSort": 1}{"SeqNumber": 82, "message": "Test Message - 83", "val3": "Value 3 - 83", "val2": "Value 2 - 83", "val1": "Value 1 - 83", "SeqSort": 1}{"SeqNumber": 83, "message": "Test Message - 84", "val3": "Value 3 - 84", "val2": "Value 2 - 84", "val1": "Value 1 - 84", "SeqSort": 1}{"SeqNumber": 84, "message": "Test Message - 85", "val3": "Value 3 - 85", "val2": "Value 2 - 85", "val1": "Value 1 - 85", "SeqSort": 1}{"SeqNumber": 85, "message": "Test Message - 86", "val3": "Value 3 - 86", "val2": "Value 2 - 86", "val1": "Value 1 - 86", "SeqSort": 1}{"SeqNumber": 86, "message": "Test Message - 87", "val3": "Value 3 - 87", "val2": "Value 2 - 87", "val1": "Value 1 - 87", "SeqSort": 1}{"SeqNumber": 87, "message": "Test Message - 88", "val3": "Value 3 - 88", "val2": "Value 2 - 88", "val1": "Value 1 - 88", "SeqSort": 1}{"SeqNumber": 88, "message": "Test Message - 89", "val3": "Value 3 - 89", "val2": "Value 2 - 89", "val1": "Value 1 - 89", "SeqSort": 1}{"SeqNumber": 89, "message": "Test Message - 90", "val3": "Value 3 - 90", "val2": "Value 2 - 90", "val1": "Value 1 - 90", "SeqSort": 1}{"SeqNumber": 90, "message": "Test Message - 91", "val3": "Value 3 - 91", "val2": "Value 2 - 91", "val1": "Value 1 - 91", "SeqSort": 1}{"SeqNumber": 91, "message": "Test Message - 92", "val3": "Value 3 - 92", "val2": "Value 2 - 92", "val1": "Value 1 - 92", "SeqSort": 1}{"SeqNumber": 92, "message": "Test Message - 93", "val3": "Value 3 - 93", "val2": "Value 2 - 93", "val1": "Value 1 - 93", "SeqSort": 1}{"SeqNumber": 93, "message": "Test Message - 94", "val3": "Value 3 - 94", "val2": "Value 2 - 94", "val1": "Value 1 - 94", "SeqSort": 1}{"SeqNumber": 94, "message": "Test Message - 95", "val3": "Value 3 - 95", "val2": "Value 2 - 95", "val1": "Value 1 - 95", "SeqSort": 1}{"SeqNumber": 95, "message": "Test Message - 96", "val3": "Value 3 - 96", "val2": "Value 2 - 96", "val1": "Value 1 - 96", "SeqSort": 1}{"SeqNumber": 96, "message": "Test Message - 97", "val3": "Value 3 - 97", "val2": "Value 2 - 97", "val1": "Value 1 - 97", "SeqSort": 1}{"SeqNumber": 97, "message": "Test Message - 98", "val3": "Value 3 - 98", "val2": "Value 2 - 98", "val1": "Value 1 - 98", "SeqSort": 1}{"SeqNumber": 98, "message": "Test Message - 99", "val3": "Value 3 - 99", "val2": "Value 2 - 99", "val1": "Value 1 - 99", "SeqSort": 1}{"SeqNumber": 99, "message": "Test Message - 100", "val3": "Value 3 - 100", "val2": "Value 2 - 100", "val1": "Value 1 - 100", "SeqSort": 1}
```

Archiviazione dati (2/2)

DynamoDB

Ho creato una table a cui è stata collegata una regola IoT per caricare tutti i dati in ingresso.

DynamoDB > Explore items: iotmqtt > Edit item

Edit item Form JSON view

You can add, remove, or edit the attributes of an item. You can nest attributes inside other attributes up to 32 levels deep. [Learn more](#)

Attributes Add new attribute ▼

Attribute name	Value	Type	
SeqNumber - Partition key	96	String	
SeqSort - Sort key	1	String	
payload	Insert a field ▼	Map	<button>Remove</button>
SeqNumber	96	Number	<button>Remove</button>
SeqSort	1	Number	<button>Remove</button>
message	Test Message - 97	String	<button>Remove</button>
val1	Value 1 - 97	String	<button>Remove</button>
val2	Value 2 - 97	String	<button>Remove</button>
val3	Value 3 - 97	String	<button>Remove</button>

Cancel Save changes

IoT rule

Per ridurre i costi ho creato una regola IoT che carica tutti i dati in arrivo dal protocollo Mqtt al Kinesis Firehose Stream attraverso un' actions.

AWS IoT > Message routing > Rules > IoTStreamMqtt

IoTStreamMqtt [Info](#) [Activate](#) [Deactivate](#) [Edit](#) [Delete](#)

Details

Description -		
ARN arn:aws:iot:eu-central-1:251714578592:rule/IoTStreamMqtt	Topic general/inbound	Created date July 12, 2023, 11:43:18 (UTC+02:00)
Status Active	Basic ingest topic \$aws/rules/IoTStreamMqtt	

SQL statement

SQL statement SELECT * FROM 'general1/inbound'	SQL version 2016-03-23
---	---------------------------

[Actions](#) | [Error action](#) | [Tags](#)

Actions (1)

Actions occur when an event is triggered. Actions are executed from top to bottom, until all actions are completed or an error occurs. To add or remove actions, you will need to edit the rule. [View details](#)

Service	Action
Kinesis Firehose stream	Send a message to an Amazon Kinesis Firehose stream

Kinesis Data Firehose

Creo una regola IoT che carica tutti i dati in arrivo dal protocollo Mqtt al Kinesis Firehose

Stream. Creando un buffer di telemetria dei messaggi ricevuti da IoT Core senza scriverli direttamente su un Bucket S3, siamo in grado di ridurre i costi dell'ingest di quasi 90%.

The screenshot shows the Amazon Kinesis Data Firehose console. The left sidebar contains navigation links: Dashboard, Data streams, Data Firehose, Analytics applications, Resources, and What's new. The main content area displays the configuration for a delivery stream named 's3-bucket-iot-pw-tag'. The 'Delivery stream details' section shows the stream is 'Active', the source is 'Direct PUT', the destination is 'Amazon S3', and the ARN is 'arn:aws:firehose:eu-central-1:251714578592:deliverystream/s3-bucket-iot-pw-tag'. It also shows that data transformation and dynamic partitioning are 'Not enabled'. The 'Test with demo data' section includes a script to simulate data and buttons to start and stop sending demo data. The bottom of the console shows tabs for Monitoring, Configuration, and Destination error logs.



Shadow device

Il suo utilizzo offre diversi vantaggi:

- **Persistenza dei dati:** Device Shadow fornisce uno stato persistente per i dispositivi IoT. Ciò significa che i dispositivi possono mantenere uno stato coerente anche se non sono costantemente connessi alla rete. Quando il dispositivo si riconnette, può sincronizzare le sue informazioni con la sua “ombra” sul cloud.
- **Gestione dei dispositivi:** AWS IoT semplifica la gestione di grandi flotte di dispositivi IoT. Possiamo monitorare, controllare e gestire in modo centralizzato tutti i tuoi dispositivi IoT attraverso la console di gestione di AWS IoT o tramite API.
- **Controllo remoto:** con Device Shadow, è possibile controllare e aggiornare lo stato dei dispositivi IoT da remoto. Inviare comandi o modificare le impostazioni del dispositivo attraverso l'ombra del dispositivo senza dover comunicare direttamente con il dispositivo fisico.

In IoT Core, abbiamo creato una "Thing" che rappresenta il nostro dispositivo IoT. Abbiamo assegnato un nome univoco alla Thing e associando il certificato e la chiave privata del dispositivo ad essa. Ho utilizzato l'API di gestione di AWS IoT Core per configurare l'ombra del dispositivo (Device Shadow). L'ombra del dispositivo è una rappresentazione virtuale del tuo dispositivo, che contiene lo stato attuale del dispositivo e le informazioni associate ad esso.

Web App per invio pub in MQTT

Di seguito i vantaggi che abbiamo riscontrato in Amplify nel mondo IoT:

Vantaggi:

- Facilità d'integrazione con altri servizi AWS: Amplify può integrarsi con una vasta gamma di servizi AWS, come AWS Lambda, Amazon DynamoDB, Amazon S3 e molti altri. Questo ci ha consentito di collegare tutti i servizi facilmente.
- Sicurezza: AWS offre un robusto set di strumenti di sicurezza per proteggere le applicazioni IoT. Amplify supporta l'autenticazione, l'autorizzazione e la crittografia dei dati per garantire che le applicazioni IoT siano protette da minacce esterne.

Svantaggi:

- Costi e dipendenza ad AWS: Dopo i primi 12 mesi gratuiti attraverso il piano free tier, il servizio diventa a pagamento ed in qualche modo ci siamo vincolati alla piattaforma, una soluzione indipendente potrebbe essere una via migliore
- Documentazione: Se per l'integrazione dell'autenticazione tramite Cognito non ho avuto problemi, per l'utilizzo della libreria Pub/Sub usando React ho trovato la documentazione ufficiale meno guidata.

In generale possiamo affermare che con l'utilizzo di Amplify sono riuscito ad esporre un applicativo in tempi più brevi rispetto a svilupparne uno partendo da zero.

Cloudwatch Alarms - SNS


Creazione degli Allarmi di errore collegati a delle Actions ed a Topic SNS per fare in modo di venire notificati in caso di problemi.

CloudWatch > Alarms

Alarms (9) ☐ Hide Auto Scaling alarms [Clear selection](#)

Any state ▼ Any type ▼ Any actions ... ▼

<input type="checkbox"/>	Name ▼	State ▼	Last state update ▼	Conditions	Actions
<input type="checkbox"/>	TargetTracking-table/iotmqtt-ProvisionedCapacityHigh-6c241813-e7de-4d3c-87c0-f39816e78dcf	OK	2023-07-17 08:39:19	ProvisionedWriteCapacityUnits > 1 for 3 datapoints within 15 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-table/iotmqtt-AlarmHigh-e6368dba-17ef-4475-b125-60d375be4cf9	OK	2023-07-17 08:35:28	ConsumedWriteCapacityUnits > 42 for 2 datapoints within 2 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-table/iotmqtt-AlarmLow-cf778429-4dd1-4b3e-b3c1-2d0c0d7bbef2	In alarm	2023-07-17 08:35:03	ConsumedWriteCapacityUnits < 30 for 15 datapoints within 15 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-table/iotmqtt-ProvisionedCapacityLow-31437af9-4d56-42a6-a079-2692eb05e89d	OK	2023-07-17 08:34:16	ProvisionedWriteCapacityUnits < 1 for 3 datapoints within 15 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-table/iotmqtt-ProvisionedCapacityHigh-9ffaf337-d6dc-40c9-9bcf-1ff82d1a2bbe	OK	2023-07-17 08:24:06	ProvisionedReadCapacityUnits > 1 for 3 datapoints within 15 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-table/iotmqtt-AlarmLow-96dc6d44-2ff9-4aaa-949e-1bb3986f4e2e	In alarm	2023-07-17 08:23:32	ConsumedReadCapacityUnits < 30 for 15 datapoints within 15 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-table/iotmqtt-AlarmHigh-facff65b-a468-431e-8062-1e0740170001	OK	2023-07-17 08:23:23	ConsumedReadCapacityUnits > 42 for 2 datapoints within 2 minutes	Actions enabled
<input type="checkbox"/>	TargetTracking-table/iotmqtt-ProvisionedCapacityLow-c960f5bf-a4ca-42e4-hzfh-z4ac70c7f7af	OK	2023-07-17 08:22:37	ProvisionedReadCapacityUnits < 1 for 3 datapoints within 15 minutes	Actions enabled

ALARM: "Iot Mqtt Device error" in EU (Frankfurt)  Posta in arrivo x



AWS Notifications

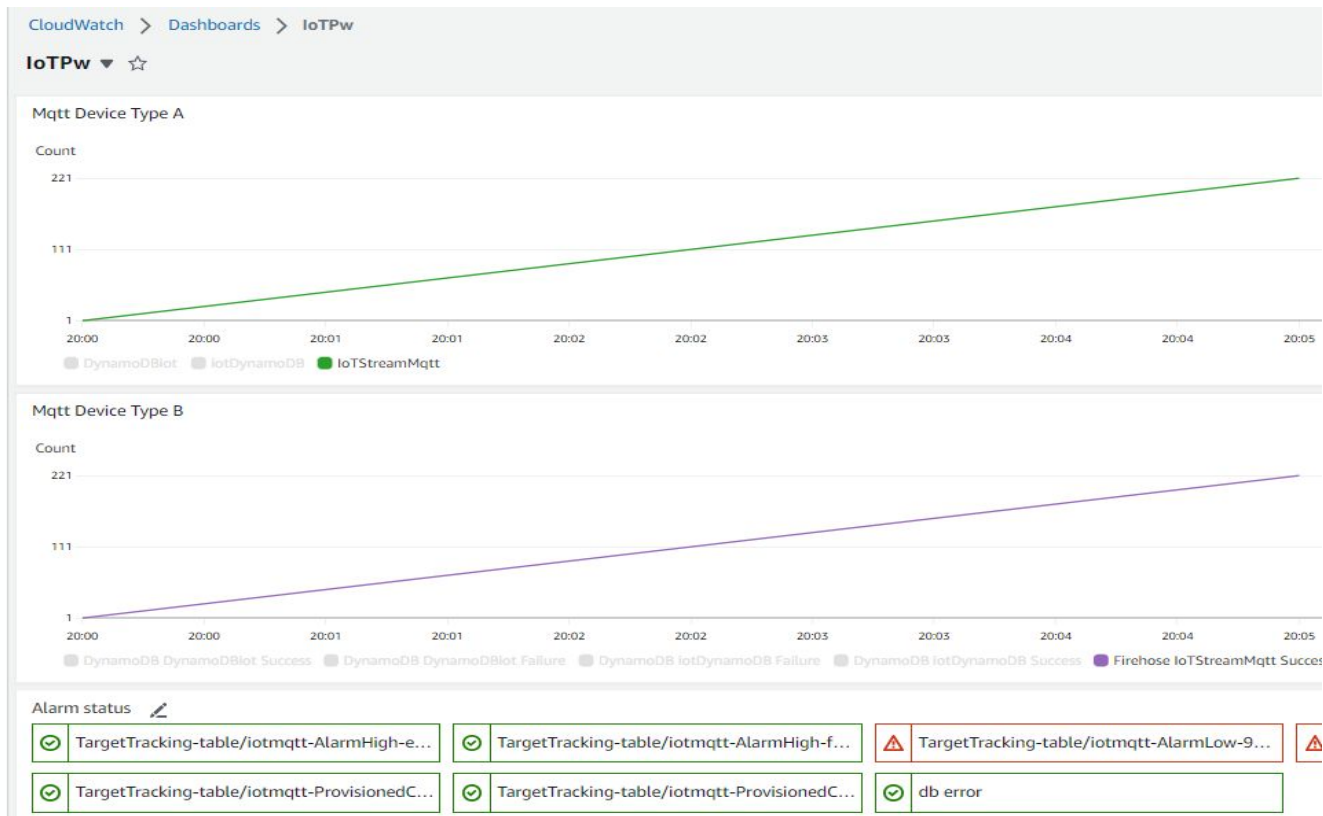
a me ▼

Cloudwatch Dashboard

Per rientrare nel free tier ho creato 3 dashboard (una per gli allarmi , una per device di tipo A ed una di tipo B)

selezionando meno di 50 metriche. La dashboard può essere visualizzata tramite link pubblico oppure per maggiore sicurezza creando un utente

IAM associando l'accesso limitato alla visualizzazione delle dashboard.



Sicurezza

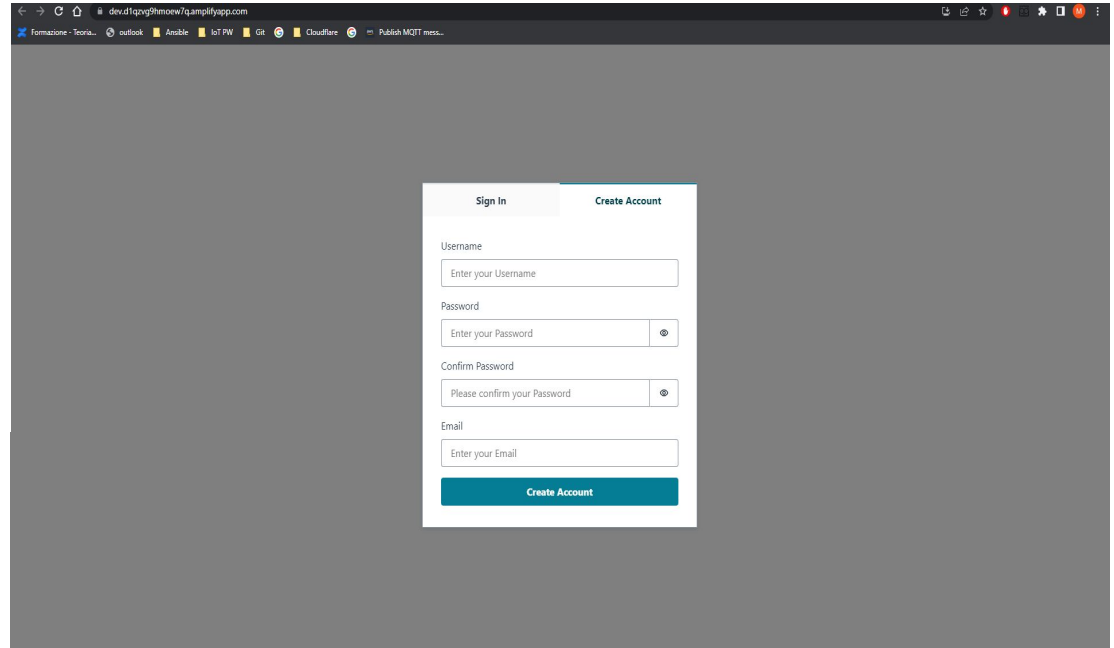
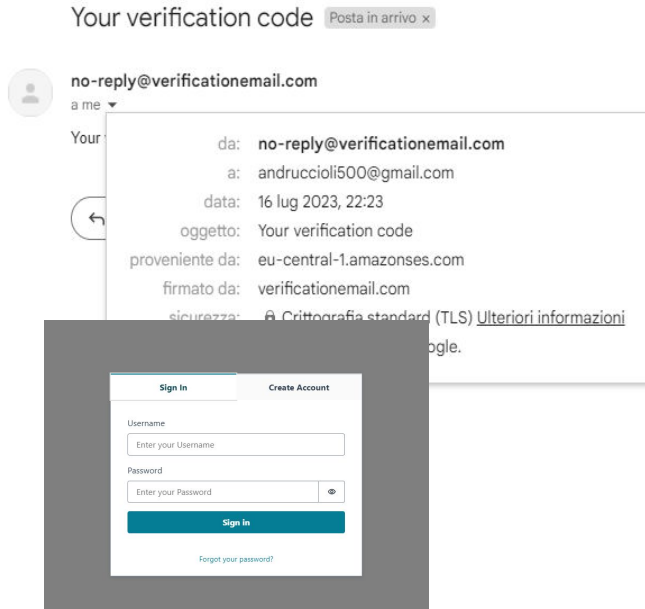
IAM-Cognito-Certificati Device-IoT Policy per garantire il livello di sicurezza più alto possibile, i permessi sono stati forniti sempre in ottica di **least privilege**:

- **IAM** consente una gestione centralizzata delle identità e dei privilegi degli utenti, garantendo che solo le persone autorizzate possano accedere alle risorse e ai servizi IoT.
- **Cognito** fornisce un'infrastruttura scalabile per l'autenticazione degli utenti, consentendo un accesso sicuro alle applicazioni IoT.
- I **certificati per dispositivi** offrono un meccanismo di autenticazione robusto tra il dispositivo IoT e i servizi cloud, proteggendo da potenziali attacchi di falsificazione o manipolazione.
- **IoT Policy** consentono di definire in modo granulare i permessi e le restrizioni di accesso ai dispositivi e ai dati IoT, garantendo la confidenzialità e l'integrità delle informazioni.

L'unione ed utilizzo di tutte le possibilità sopra contribuisce a mitigare i rischi di sicurezza nell'ambito dell'IoT e a garantire un ambiente protetto per le applicazioni e i dispositivi.

Amplify - Cognito -IAM (1/3)

Autenticazione usando Amplify-Cognito-IAM in React, hostata gratuitamente su AWS



Amplify - PubSub (2/3)

Una volta autenticato, se l'utente dispone delle autorizzazioni Cognito-IAM necessarie, può inviare comandi ad un device attraverso lo specifico topic di riferimento.

La nostra ec2 che è iscritta al topic riceve il messaggio e tramite una condizione effettua l'azione

Project Work iot
utente loggato:
andruccioli500

Sign out

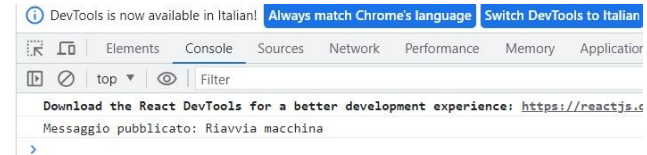
MQTT TOPIC

device1

Riavvia macchina

Publish Message

Messaggio pubblicato



Riavvio macchina in corso....

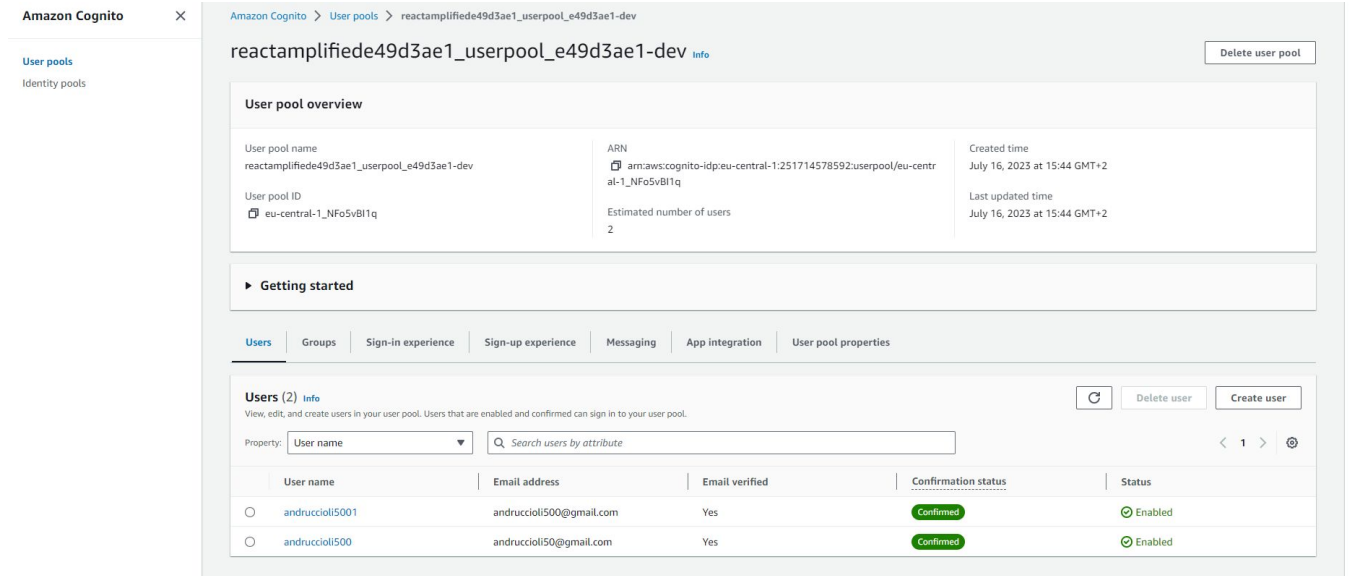
Macchina riavviata

ubuntu:~/environment \$

Amplify - Cognito (3/3)

Per questioni di comodità e sicurezza, si potrebbe cancellare la parte di registrazione lasciando solo quella di login.

La creazione degli account abilitati avverrebbe in maniera manuale attraverso User Pool di Cognito.



The screenshot shows the Amazon Cognito console interface. The left sidebar contains the navigation menu with 'User pools' selected. The main content area displays the details for the user pool 'reactamplifiede49d3ae1_userpool_e49d3ae1-dev'. Below the overview section, the 'Users' tab is active, showing a list of users. The table has columns for User name, Email address, Email verified, Confirmation status, and Status. Two users are listed, both with a 'Confirmed' status and 'Enabled' status.

User name	Email address	Email verified	Confirmation status	Status
andruccioli5001	andruccioli500@gmail.com	Yes	Confirmed	Enabled
andruccioli500	andruccioli50@gmail.com	Yes	Confirmed	Enabled

Scalabilità e resilienza

La scalabilità e la resilienza sono aspetti critici per garantire che il sistema possa gestire l'aumento del carico di lavoro, rimanere affidabile anche in caso di guasti e garantire una disponibilità continua dei servizi. AWS offre soluzioni per affrontare queste sfide nel tuo progetto.

Autoscaling:

- AWS fornisce strumenti per la scalabilità automatica dei servizi, consentendo di adattare le risorse alle esigenze del carico di lavoro.
- L'utilizzo di servizi come Amazon EC2 Auto Scaling per scalare automaticamente le istanze dei server in base alla domanda. Questo consente di gestire l'aumento o la diminuzione del traffico senza dover gestire manualmente l'infrastruttura.

Distribuzione multi-zona e multi-regione:

- AWS consente di distribuire il sistema IoT su più zone di disponibilità all'interno di una regione o su diverse regioni geografiche.
- La distribuzione multi-zona e multi-regione aumenta la resilienza del sistema, consentendo di gestire guasti a livello di zona o di intere regioni senza interrompere l'operatività.
- Utilizzare servizi come Amazon Route 53 per la gestione del traffico e la ridondanza geografica garantisce un'alta disponibilità del sistema.

Fault-tolerance e Disaster Recovery:

- AWS offre servizi che consentono di progettare soluzioni fault-tolerant e di ripristino di emergenza nel caso di eventi catastrofici o guasti hardware.
- Utilizzare servizi come Amazon S3 garantisce un'archiviazione dei dati con durabilità elevata e la replica automatica dei dati in diverse zone di disponibilità.

Costi

I costi sono un aspetto cruciale che ho considerato durante il PW, e utilizzare lo strumento AWS Pricing Calculator mi ha aiutato a valutare e pianificare i costi del nostro progetto IoT basato su AWS.

Attraverso il calcolatore, ho analizzato i costi previsti per i servizi utilizzati nella nostra architettura, tenendo conto di fattori come il numero di dispositivi IoT, il volume dei dati, la frequenza di aggiornamento e l'utilizzo dei servizi AWS.

Grazie a questa analisi preventiva, sono stato in grado di pianificare e gestire il budget del progetto in modo efficace. Identificato i costi principali, ad esempio quelli associati all'utilizzo di AWS IoT Core, Amazon S3, Amazon DynamoDB e altri servizi pertinenti. Inoltre, abbiamo considerato il modello di pagamento basato sul consumo effettivo di AWS, che ci ha permesso di adattare le risorse in base alle necessità del progetto e di evitare sprechi.

Posso confermare che, grazie a una pianificazione accurata e ad un utilizzo oculato delle risorse, sono rimasti all'interno del budget stabilito per il progetto. Questo significa che abbiamo gestito i costi in modo efficace, senza superare le aspettative finanziarie e assicurando una gestione finanziaria responsabile del progetto IoT.

Tenendo conto dei costi e del budget, abbiamo mantenuto un equilibrio tra l'implementazione di una soluzione robusta e performante e la gestione dei costi operativi.

Per la parte costi vedi file <https://docs.google.com/document/d/1HvHallkiL7yi-omjxAv0yAvOYCIUOE76c7LgpNBZHDk/edit>

Risultati e vantaggi

Utilizzando l'architettura proposta basata su AWS nel nostro PW, i risultati e i vantaggi includono:

- Monitoraggio in tempo reale per rilevare anomalie e prendere decisioni tempestive.
- Ottimizzazione delle prestazioni tramite l'elaborazione dei dati in tempo reale.
- Archiviazione scalabile dei dati e analisi storica per identificare pattern e tendenze.
- Sicurezza avanzata con autenticazione e autorizzazione per i dispositivi IoT.
- Scalabilità automatica e riduzione dei costi grazie all'elasticità dei servizi AWS.

Complessivamente, ho ottenuto una maggiore efficienza operativa, decisioni informate, protezione dei dati e un'architettura adattabile che può gestire l'espansione del progetto a costi ottimizzati.

Soluzioni alternative e sviluppi futuri

Per quanto riguarda possibili sviluppi futuri, ho pensato all'implementazione di:

- GreenGrass, quicksight , aws glue e Athena
- Manutenzione predittiva tramite l'uso di **SageMaker**, impiegando modelli di Machine Learning per il riconoscimento di potenziali guasti dell'attrezzatura e il suggerimento di azioni per la loro risoluzione: un'istanza del notebook di SageMaker viene utilizzata per orchestrare il modello e un'altra istanza per eseguire l'addestramento. Il codice di addestramento e il modello addestrato vengono memorizzati nel bucket Amazon S3 della soluzione. **Vantaggi:** consente di creare e addestrare rapidamente e facilmente modelli di machine learning e di distribuirli direttamente in un ambiente di produzione pronto all'uso.



Elaborazione dei dati - alternativa

Per l'elaborazione dei dati provenienti dai dispositivi IoT, come soluzione alternativa trovata è stata quella di utilizzare AWS Lambda, quest'ultimo è un servizio di elaborazione server-less che ci consente di eseguire il codice in risposta agli eventi, come l'arrivo di nuovi dati dai sensori.

Di seguito come abbiamo utilizzato Lambda per l'elaborazione dei dati IoT nel nostro progetto:

1. Creazione di una funzione Lambda: abbiamo creato una funzione Lambda personalizzata che definisce il codice da eseguire sull'arrivo dei dati dai sensori. Questo codice include trasformazioni, analisi e operazioni specifiche
2. Configurazione del trigger: Abbiamo configurato un trigger della funzione Lambda per essere attivato quando nuovi dati arrivano da AWS IoT Core. In questo modo, ogni volta che un dispositivo invia dati al cloud, la funzione Lambda viene automaticamente attivata per elaborare quei dati.
3. Elaborazione dei dati: All'interno della funzione Lambda, abbiamo scritto un codice personalizzato per elaborare i dati provenienti dai sensori. Ciò ci è servito anche per normalizzare i dati, trasformarli in un formato specifico o filtrarli per rimuovere informazioni non necessarie.
4. Integrazione con altri servizi: La funzione Lambda è stata integrata con altri servizi di AWS per ulteriori elaborazioni o azioni. Ad esempio, abbiamo inviato i dati elaborati a DynamoDB per l'archiviazione a lungo termine.

I vantaggi nell'utilizzo di AWS Lambda includono:

1. Scalabilità automatica: Lambda gestisce automaticamente la scalabilità delle nostre funzioni in base al volume dei dati e al carico di lavoro. Non è necessario preoccuparsi di dimensionare o gestire l'infrastruttura sottostante, poiché Lambda si adatta automaticamente alla richiesta di elaborazione.
2. Riduzione dei costi: Con Lambda il "pagamento in base al consumo" è stato vantaggioso in termini di riduzione dei costi.
3. Velocità di elaborazione: Lambda offre una risposta rapida agli eventi, consentendo l'elaborazione in tempo reale dei dati provenienti dai sensori.
4. Integrazione con il resto dell'ecosistema di AWS: Lambda si integra facilmente con altri servizi di AWS, consentendo di creare pipeline di dati e sfruttare funzionalità avanzate come l'archiviazione, l'analisi o la generazione di notifiche in modo semplice e coerente.

Conclusioni

In conclusione, la nostra architettura basata su AWS rappresenta, una soluzione altamente efficace per affrontare le sfide del progetto. Ecco alcuni punti chiave che evidenziano la concretezza e l'efficacia della nostra architettura:

1. **Affidabilità:** Grazie alla distribuzione su più zone di disponibilità e regioni geografiche di AWS, il sistema IoT è altamente resiliente ai guasti e alle interruzioni. Ciò assicura che i dispositivi IoT rimangano operativi e che i dati vengano raccolti e trasmessi in modo affidabile.
2. **Scalabilità:** Utilizzando servizi come Amazon EC2 Auto Scaling, siamo in grado di adattare le risorse del sistema in base al carico di lavoro in tempo reale. Ciò garantisce che il sistema possa gestire picchi di traffico e aumenti del numero di dispositivi IoT senza compromettere le prestazioni.
3. **Sicurezza:** AWS fornisce una vasta gamma di strumenti e servizi per garantire la sicurezza dei dispositivi IoT e dei dati. L'utilizzo di AWS IoT Core consente autenticazione, autorizzazione e crittografia dei dati, proteggendo il sistema da minacce e violazioni della sicurezza.
4. **Gestione semplificata:** Grazie a servizi come AWS IoT Core, la gestione del sistema IoT diventa semplice ed efficiente. Possiamo monitorare le prestazioni, configurare i dispositivi e ricevere notifiche in tempo reale, semplificando l'amministrazione e il monitoraggio del sistema.
5. **Costi ottimizzati:** Utilizzando il modello di pagamento basato sul consumo effettivo di AWS, possiamo ottimizzare i costi operativi. Paghiamo solo per le risorse che effettivamente utilizziamo, evitando sprechi e garantendo un utilizzo efficiente delle risorse.

In definitiva, l'architettura offre una soluzione concreta e affidabile per il progetto IoT. Grazie alla sua scalabilità, sicurezza, gestione semplificata e costi ottimizzati, crediamo di aver raggiunto gli obiettivi del PW in modo efficiente, fornendo un sistema robusto e performante per la raccolta, l'elaborazione e la visualizzazione dei dati provenienti dai dispositivi.