

Istruzioni per Configurazione ed esecuzione del sistema

Ludovico Di Iorio | 0336019 | ludovico.diiorio@students.uniroma2.eu

Il primo passaggio essenziale per poter far funzionare l'intero sistema è quello di avviare le istanze EC2 di AWS, e dopo qualche secondo il servizio sarà disponibile e online.

Admin

Per l'amministratore del sistema è disponibile un eseguibile (admin.exe) che consente di eseguire diverse operazioni, come viene riportato in figura:



- 1) e 2) eseguono uno query sul db per mostrare tutti i sensori e i parchi registrati nel sistema, sia attivi che non attivi.



• 3) serve per aggiungere un sensore, come nel seguente caso e sarà poi visualizzabile con il comando 1)



Da notare che, quando un sensore viene aggiunto, non essendo associato ad alcun parco, inizialmente sul campo park id viene assegnato il valore '0'

• 4) serve per aggiungere un parco, come nel seguente caso e sarà poi visualizzabile con il comando 2)

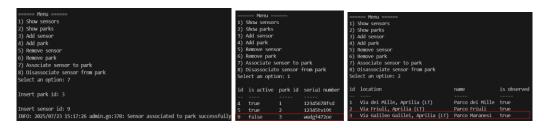


Da notare che, all'inserimento di un nuovo parco, il campo is_observed è posto a false per indicare che non c'è alcun sensore che è stato ancora assegnato al parco.

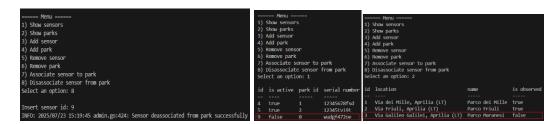
• I comandi 5) e 6) consentono permettono di rimuovere rispettivamente i sensori o i parchi, però per poter essere eliminati è necessario che il sensore non si associato ad alcun parco e che il parco non sia associato ad alcun sensore.

```
| Menu |
```

• Il comando 7) serve per associare un sensore ad un parco e sarà sufficiente inserire l'id del sensore e poi l'id del parco a cui lo si vuole assegnare.



• Con 8) si può disassociare un sensore dal parco a cui è stato assegnato, ed è sufficiente inserire l'id del sensore.



Tutte queste funzioni sono eseguite come transazioni, le quali consentono di evitare problemi di inconsistenza in caso di errori o problemi nell'esecuzione.

Edge

Per la creazione e l'esecuzione di un dispositivo edge si seguono i seguenti passaggi:

- 1. Essendo un dispositivo simulato, creiamo un'istanza EC2 su AWS, sulla quale dovremo installare k3s (curl -sfL https://get.k3s.io | sh -) e poi avviamo il servizio (sudo systemctl start k3s).
- 2. Ora dobbiamo definire i file Manifest che serviranno per la creazione del pod, sul quale girerà il programma:
 - a. edge-cm.yaml: file di tipo ConfigMap, che viene sfruttato per definire le variabili d'ambiente, e in questo caso l'unica sarà SERIAL_NUMBER, che contiene il numero seriale del dispositivo.

```
apiVersion: v1
kind: ConfigMap
metadata:
    name: edge-cm
data:
    SERIAL_NUMBER: "wudgf472oe"
```

b. edge.yaml: file di tipo Deployment che serve per creare il pod, quindi il container sul quale girerà il microservizio sviluppato in Python

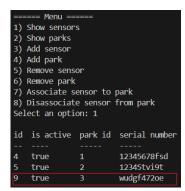
```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: edge-deployment
labels:
    app: edge
  replicas: 1
  selector:
    matchLabels:
  app: edge
template:
     metadata:
       labels:
         app: edge
     spec:
       containers:
         name: edge
         image: ludovicodiiorio/edge-repo:0.05
         envFrom:
          - configMapRef:
              name: edge-cm
```

Con questo file si dichiara il numero di pod che si vogliono creare (1 pod), la repository di Docker Hub dalla quale viene prelevata l'immagine del container e infine c'è un riferimento al file edge-cm.yaml dal quale sono recuperate quelle che saranno le variabili d'ambiente.

- 3. Per applicare i file manifest e quindi creare i componenti è sufficiente usare i comandi *sudo kubectl apply -f edge-cm.yaml* e *sudo kubectl apply -f edge.yaml*, ma nel seguente ordine poiché nel secondo file manifest viene fatto riferimento al primo, quello del Config Map.
- 4. Adesso possiamo verificare se l'edge è partito ed è funzionante con il comando *sudo kubectl get pod* con il quale potremo verificare se il pod è stato creato, se è partito ed è in salute:

```
[ec2-user@ip-172-31-87-70 k3s]$ sudo kubectl get pod
NAME READY STATUS RESTARTS AGE
edge-deployment-54f757dcf6-bcs7j 1/1 Running 0 3m2s
```

A questo punto l'admin può confermare che il dispositivo è attivo nel parco e sta raccogliendo misurazioni, andando a visualizzare il parametro *is_active* che ora dovrebbe essere *true*.



Una maniera ancora più semplice per verificare se il dispositivo edge sta funzionando è quello di andare a verificare sulla pagina web, sulla quale ci dovrebbe essere il nuovo parco tra quelli osservati



Frontend & Backend

I microservizi per il backend e per il frontend sono in esecuzione sulla stessa istanza EC2, e si trovano nel cluster Minikube. La loro esecuzione è stata automatizzata, quindi all'avvio dell'istanza si avvia il cluster ed anche i pod: ne avremo uno per il microservizio dedicato alla comunicazione con i dispositivi edge, uno dedicato ad esporre le API e poi ne avremo due per il server Nginx (quindi una replica) per aumentare la disponibilità del servizio web.

```
[ec2-user@ip-172-31-9-233 ~]$ kubectl get pod
NAME
                                           READY
                                                   STATUS
                                                              RESTARTS
                                                                                AGE
api-server-deployment-78d4dc4f6d-s75mx
                                                   Running
                                                              13 (4d23h ago)
                                                                                6d3h
                                           1/1
                                           1/1
1/1
backend-deployment-649bd84d68-7qnlz
                                                   Running
                                                              13
                                                                 (4d23h ago)
                                                                                6d3h
                                                                 (4d23h ago)
                                                                                6d3h
frontend-deployment-7ddc48546b-4hpsr
                                                   Running
                                                              13
frontend-deployment-7ddc48546b-dqt2h
                                                   Running
                                                                 (4d23h ago)
                                                                                6d3h
[ec2-user@ip-172-31-9-233 ~]$
```

Nello script che automatizza l'avvio del cluster, sono state inserite anche delle istruzioni per garantire che i servizi siano raggiungibili e che in caso di fallimenti siano ripristinati:

- Con il comando *kubectl port-forward* le porte dell'host (l'istanza EC2) sono mappate con quelle del container di Minikube (essendo Docker il driver su cui si appoggia Minikube, esso è eseguito come un container), cosicché le richieste in arrivo all'istanza siano reindirizzate a minikube, e quindi ai pod.
- Sono state inserite nello script anche degli accorgimenti per far sì che in caso di problemi il cluster sia riavviato e le mappature tra le porte dell'host e quelle di minikube siano ristabilite.

È importante sapere che i componenti Service associati ai pod, sono di tipo Nodeport, ovvero servono per esporre un servizio all'accesso esterno aprendo una porta specifica su tutti i nodi del cluster; per deafult, però le porte del nodo assegnabili stanno in un range limitato (30000-

32767), per cui sono state fatte le seguenti assegnazioni per i servizi che girano nel cluter di minikube:

servizio	Porta dell'host	Porta del pod
Web server	32009	80
API server	31003	8080
gRPC server	31008	50051

Di conseguenza per poter usufruire del servizio web, sulla barra dell'URL del browser sarà necessario scrivere: 44.214.125.195:32009.

N.B. le istanze EC2 di AWS ogni volta che vengono avviate, viene assegnato loro un nuovo indirizzo IP, e ciò può dare problemi nel momento in cui assumono il ruolo di server. Per risolvere tale problema, all'istanza che contiene il cluster e quella col database è stato assegnato un indirizzo IP statico, grazie al servizio Elastic IP di AWS.

Database

Per l'avvio del database, il processo è lo stesso usato per i servizi di backend e frontend, per cui è sufficiente avviare l'istanza EC2 sulla quale gira il servizio, e dopodiché sarà possibile interagire col database.