

How To

Luca Mastrobattista, 0292461

1 Consumer

Nella cartella `src/consumer` sono presenti due script *bash* che hanno lo scopo di inizializzare e distruggere l'infrastruttura cloud.

1.1 `creator.sh`

Permette la creazione dell'infrastruttura tramite il file *Terraform* `src/consumer/init_infrastructure/infrastructure.tf`. Questo script può ricevere due parametri di input:

- `-env_var`: questo parametro specifica di non usare il file `~/.aws/credentials` per leggere le credenziali di accesso, ma anzi vengono richieste a riga comando per essere settate come variabili di ambiente;
- `-no_db`: questo parametro specifica di non re-inizializzare il database. In realtà, è maggiormente utile in fase di sviluppo, quando si effettua un aggiornamento sul codice e si vuole mantenere il database intatto.

Lanciare lo script senza parametri porta alla creazione di un ambiente cloud *pulito*, prendendo le credenziali dal file `~/.aws/credentials`.

1.2 `destroyer.sh`

Distrugge l'infrastruttura creata precedentemente. Anche questo script, può ricevere in input il parametro opzionale `-env_var`, che ha lo stesso scopo e produce lo stesso comportamento del caso precedente.

2 Producer

Il client può essere lanciato in due modalità differenti:

- `$ python main.py`: il client viene lanciato con un'interfaccia CLI.
- `$ python main.py -g`: il client viene lanciato con il supporto GUI.

Anche qui, è necessario specificare dove debbano essere recuperate le credenziali di accesso per interagire con i servizi AWS.

Se si sceglie di utilizzare il file `~/.aws/credentials`, dopo averlo correttamente configurato, si può procedere al lancio dell'applicazione, altrimenti è necessario lanciare lo script bash `configure.sh` che chiederà interattivamente le credenziali e le memorizzerà su un file nascosto chiamato `.env`. Questo file verrà usato dall'applicazione per impostare le variabili d'ambiente grazie alla libreria `decouple`.

Nota: non è previsto un meccanismo di eliminazione automatica del file `.env`