

Documentation du Load Balancer en Rust

Introduction

Ce projet est un load balancer simple écrit en Rust. Il utilise hyper pour le serveur HTTP et tokio pour l'asynchronicité. Le load balancer distribue les requêtes entrantes entre deux serveurs backend basés sur leur santé et disponibilité.

Fonctionnalité

Le load balancer surveille la santé de deux serveurs backend simples. Il redirige les requêtes entrantes vers le backend sain. Si les deux backends sont considérés comme non sains, le load balancer répondra directement avec une erreur.

Comment ça fonctionne ?

Architecture

Le projet est composé de trois composantes principales :

Load Balancer ([main.rs](#)): Point d'entrée principal qui accepte les requêtes entrantes et les redirige vers les backends sains.

Backend Servers ([backends1.rs](#) et [backends2.rs](#)): Serveurs simples qui répondent aux requêtes reçues du load balancer.

Health Checker ([backends.rs](#)): Vérifie périodiquement la santé des backends et met à jour leur état dans le système.

Processus de Décision

Réception de la Requête : Le load balancer reçoit une requête sur le port 8080.

Sélection du Backend : Il consulte l'état de santé des backends et sélectionne un backend sain.

Redirection de la Requête : La requête est redirigée vers le backend choisi. Si aucun backend n'est sain, une erreur est renvoyée.

Boucle de Vérification de Santé

La boucle de vérification de santé exécutée en arrière-plan vérifie la santé de chaque backend toutes les 30 secondes. Un backend est considéré comme sain si une requête HTTP GET reçoit une réponse de succès.

Configuration

Dépendances : hyper, tokio, lazy_static, futures.

Ports : Le load balancer écoute sur le port 8080. Les backends écoutent sur les ports 3000 et 3001.

Guide de Démarrage

Installation des Dépendances : Exécutez cargo build pour installer les dépendances requises.

Lancement : Exécutez cargo run --bin loadbalancer pour démarrer le load balancer.

Démarrage des Backends : Dans des terminaux séparés, exécutez cargo run --bin backends1 et cargo run --bin backends2.

Tests

Lancer les Serveurs Backend

Pour le backend 1

```
1 | cargo run --bin backends1
```

Pour le backend 2 :

```
1 | cargo run --bin backends2
```

Démarrer le loadbalancer

```
1 | cargo run --bin loadbalancer
```

Simuler la Défaillance d'un Serveur Backend

Pour simuler une défaillance d'un serveur backend et observer comment le load balancer réagit, vous pouvez simplement arrêter un des serveurs backend. Pour ce faire, retournez dans le terminal où le serveur backend que vous souhaitez "faire échouer" est en cours d'exécution et utilisez Ctrl+C pour l'arrêter.

Envoyer une Requête au Load Balancer

```
1 | curl http://127.0.0.1:8080
```