

# Technical Design Document

**Projet** : Application Ruby on Rails pour les abonnés au blog

**Auteur** : Florent, Victoire, Yann, Ousmane

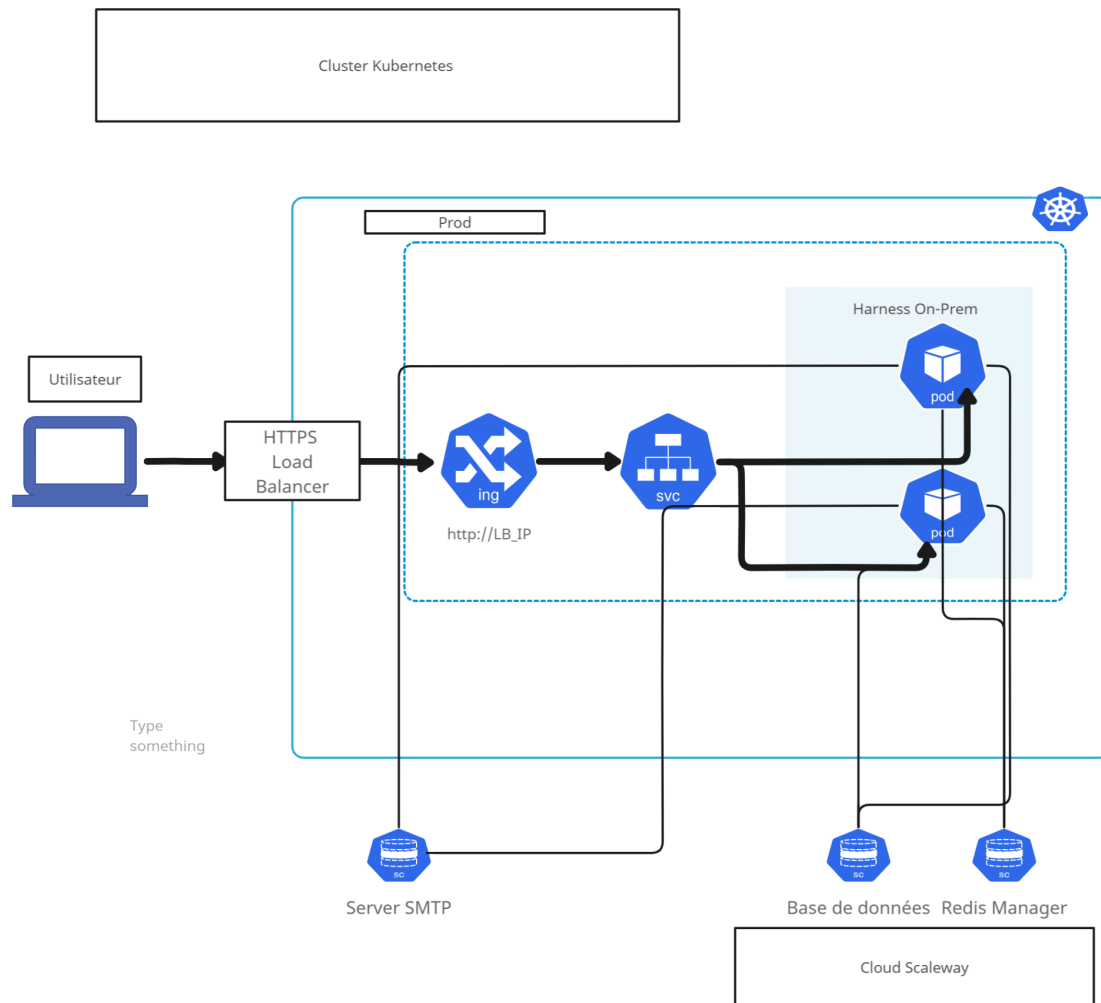
**Date** : 26 mai 2023

## 1. Introduction

Ce document décrit la conception technique pour le déploiement d'une application Ruby on Rails pour les abonnés au blog, en utilisant un cluster Kubernetes sur Scaleway. L'application comprendra deux déploiements : un pour le projet et un pour le worker. Les deux déploiements seront connectés à une base de données MySQL sur Scaleway, une base de données Redis sur Scaleway et un serveur SMTP SendGrid. Un contrôleur Ingress sera utilisé pour exposer l'application à Internet.

## 2. Architecture

Le diagramme suivant illustre l'architecture du déploiement proposé :



L'application sera déployée sur un cluster Kubernetes sur Scaleway.

L'application comprendra deux déploiements : un pour le projet et un pour le worker. Le déploiement du projet contiendra le code et les données de l'application. Le déploiement du worker contiendra le code de l'application avec un exécution de Resque.

L'application sera connectée à une base de données MySQL sur Scaleway. La base de données MySQL stockera les données de l'application.

L'application sera connectée à une base de données Redis sur Scaleway. La base de données Redis stockera les données de session de l'application.

L'application sera connectée à un serveur SMTP SendGrid. Le serveur SMTP SendGrid sera utilisé pour envoyer des notifications par e-mail.

Un contrôleur Ingress sera utilisé pour exposer l'application à Internet. Le contrôleur Ingress dirigera le trafic vers les pods de l'application en fonction du chemin d'URL.

### 3. Composants

Les composants suivants seront utilisés dans le déploiement proposé :

- Cluster Kubernetes sur Scaleway
- Base de données MySQL sur Scaleway
- Base de données Redis sur Scaleway
- Serveur SMTP SendGrid
- Contrôleur Ingress

### 4. Déploiement

Les étapes suivantes seront utilisées pour déployer l'application :

- Créer un cluster Kubernetes sur Scaleway.
- Créer une base de données MySQL sur Scaleway.
- Créer une base de données Redis sur Scaleway.
- Créer un compte SMTP SendGrid.
- Installer le contrôleur Ingress.
- Déployer l'application.

### 5. Exploitation et maintenance

Surveiller l'état de l'application depuis l'interface scaleway et optimiser les ressources allouées en fonction de la scalabilité du système. par default chaque pod à comme config :

```
resources:
  requests:
    cpu: 100m
    memory: 500M
  limits:
    cpu: 100m
    memory: 500M
```

### 6. Sécurité

Les mesures de sécurité suivantes seront mises en œuvre :

- Déployer les instances sur un réseau privé Scaleway.
- Utiliser les secret pour les variables d'environnement kubernetes

## 7. Conclusion

Le déploiement proposé fournira une plateforme évolutive et fiable pour héberger l'application Ruby on Rails pour les abonnés au blog. L'application sera protégée contre les menaces de sécurité et sera facile à exploiter et à maintenir.

## 8. Annexe

L'annexe suivante contient des informations supplémentaires sur le déploiement proposé :

- Cluster Kubernetes sur Scaleway
  - Un cluster Kubernetes sur Scaleway peut être créé à l'aide de la console Scaleway ou de l'API Scaleway.
- Base de données MySQL sur Scaleway
  - Une base de données MySQL sur Scaleway peut être créée à l'aide de la console Scaleway ou de l'API Scaleway.
- Base de données Redis sur Scaleway
  - Une base de données Redis sur Scaleway peut être créée à l'aide de la console Scaleway ou de l'API Scaleway.
- Serveur SMTP SendGrid
  - Un compte SMTP SendGrid peut être créé en utilisant le site web SendGrid.
- Contrôleur Ingress
  - Un contrôleur Ingress peut être installé à l'aide du package Helm de Kubernetes ou le Easy Deploy de Scaleway. Le contrôleur Ingress dirigera le trafic vers les pods de l'application en fonction du chemin d'URL.