

# Mise en Place d'un Runner Gitlab

---

Introduction .....	2
Architecture de nos runners.....	2
Mise en place .....	2
Le Docker Compose .....	3

## Introduction

Dans le cadre de notre projet **T-DEV-700**, nous avons besoin d'une machine qui exécute nos pipelines de test et de déploiement. Pour mettre en place ce "Runner" nous devons utiliser gitlab-runner qui permet d'interfacer un serveur et l'instance Gitlab via un simple token d'identification.

## Architecture de nos runners

Pour la simplicité de déploiement du runner, nous avons choisi de le déployer à l'aide de Docker, après avoir rédigé un docker compose, un runner supplémentaire ou de remplacement peut être ajouté très rapidement, cela peut aussi permettre de lancer le docker sur nos machines quand on a un plus gros besoin pour exécuter + de tests en même temps. Nous avons aussi un runner qui reste disponible tout le temps sur un serveur auto-hébergé.

Dans le [docker compose](#), il y a 3 services. Un qui se base sur docker:dind qui sera le service isolé qui créera les conteneurs dans nos pipelines. Un autre qui est le runner en lui-même, il sera connecté à l'instance Gitlab pour écouter les exécutions de pipeline, l'exécuteur utilisé est Docker, cela permet d'avoir un environnement propre à chaque pipeline. Le dernier est un service temporaire qui se charge uniquement de créer le fichier de configuration du runner et qui s'éteint juste après.

## Mise en place

L'environnement choisi pour notre runner nous permet de très facilement en créer un nouveau. Voici la marche à suivre.

Prérequis : avoir Docker installé.

Via l'interface Gitlab, créer un nouveau runner (Dans Settings -> CI/CD -> Runners), récupérer le token.

Après avoir clone le projet, mettre à jour l'url de l'instance gitlab et le token. Ensuite, utiliser la commande :

```
docker-compose -f docker-compose.runner.yml up -d
```

Le runner est déjà fonctionnel et connecté à l'instance.

# Le Docker Compose

```
version: "3.5"

services:

  dind:
    image: docker:20-dind
    restart: always
    privileged: true
    environment:
      DOCKER_TLS_CERTDIR: ""
    command:
      - --storage-driver=overlay2

  runner:
    restart: always
    image: registry.gitlab.com/gitlab-org/gitlab-runner:alpine
    depends_on:
      - dind
    environment:
      - DOCKER_HOST=tcp://dind:2375
    volumes:
      - ./config/etc/gitlab-runner:z
      - /var/run/docker.sock:/var/run/docker.sock
      - ./docker-credentials:/root/.docker

  register-runner:
    restart: 'no'
    image: registry.gitlab.com/gitlab-org/gitlab-runner:alpine
    depends_on:
      - dind
    environment:
      - CI_SERVER_URL={{gitlab_instance_url}} # À remplacer
      - REGISTRATION_TOKEN={{token}} # À remplacer
    command:
      - register
      - --non-interactive
      - --locked=false
      - --name=dedicated
      - --executor=docker
      - --docker-image=docker:20-dind
      - --docker-volumes=/var/run/docker.sock:/var/run/docker.sock
    volumes:
```

```
- ./config:/etc/gitlab-runner:z
```