

# Installation



## Fichier docker-compose

Dans cette procédure, nous allons utiliser **Docker**. Une documentation préalable à ce sujet est disponible [ici](#).

Commençons par créer un fichier `docker-compose.yaml`, adapté à nos besoins :

```
services:
  # Service de base de données PostgreSQL pour Keycloak
  keycloak-db:
    image: postgres:16 # Utilise l'image officielle PostgreSQL version 16
    container_name: keycloak-db
    environment:
      POSTGRES_DB: keycloak # Nom de la base de données
      POSTGRES_USER: keycloak # Nom de l'utilisateur
      POSTGRES_PASSWORD: keycloak # Mot de passe de l'utilisateur
    volumes:
      - keycloak-db-data:/var/lib/postgresql/data # Persistence des données de la base
    networks:
      - proxy # Connecté au réseau "proxy" (externe, voir plus bas)

# Service Keycloak (serveur d'identité)
keycloak:
  image: quay.io/keycloak/keycloak:24.0 # Image officielle Keycloak version 24.0
  container_name: keycloak
  command: >
    start
    --hostname=sso.domaine.com # Nom d'hôte public visible par les utilisateurs
```

```

    --hostname-strict=false                # Autorise des hôtes alternatifs (utile avec des
reverse proxy)
    --proxy-headers=xforwarded            # Interprète les headers X-Forwarded-* du proxy
(option héritée)
    environment:
        KC_DB: postgres                    # Type de base de données
        KC_DB_URL: jdbc:postgresql://keycloak-db:5432/keycloak # URL JDBC vers la base
PostgreSQL
        KC_DB_USERNAME: keycloak          # Identifiant pour se connecter à la BDD
        KC_DB_PASSWORD: keycloak          # Mot de passe pour se connecter à la BDD
        KC_HOSTNAME_STRICT: "false"       # (Redondant avec l'option de commande)
        KC_PROXY: "edge"                  # Mode proxy recommandé pour les reverse proxies
        KEYCLOAK_ADMIN: admin             # Nom de l'administrateur initial
        KEYCLOAK_ADMIN_PASSWORD: admin    # Mot de passe de l'administrateur
    depends_on:
        - keycloak-db                     # Attend que la base de données soit disponible
    ports:
        - 8080:8080                       # Expose le port 8080 (interne) sur le port 800
(hôte)
    networks:
        - proxy                            # Connecté au réseau externe "proxy"
    restart: unless-stopped               # Redémarre le service sauf si l'arrêt est manuel

# Volume Docker pour la persistance des données PostgreSQL
volumes:
    keycloak-db-data:

# Réseau Docker externe utilisé pour connecter ce service à un reverse proxy (comme Traefik ou
Apache/Nginx)
networks:
    proxy:
        external: true # Le réseau "proxy" doit déjà exister (créé manuellement)

```

# Démarrage

Une fois le fichier `docker-compose.yaml` créé, nous pouvons **lancer Keycloak** avec la commande suivante :

```
docker compose up -d
```

---

Revision #1

Created 2025-10-29 12:43:26 UTC

Updated 2025-10-29 12:43:26 UTC