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:
  mealie:
    image: ghcr.io/mealie-recipes/mealie:latest  # Utilise la dernière image officielle de
Mealie (depuis GitHub Container Registry).
    container_name: Mealie                       # Nom du container, pratique pour le
retrouver facilement dans Docker/Portainer.
    restart: always                              # Redémarre automatiquement le container en
cas de crash ou au redémarrage du système.
```

```
    ports:
        - "9925:9000"                           # Le port interne 9000 (de Mealie) est exposé sur le port 9925 de la machine hôte.
    deploy:
      resources:
        limits:
          memory: 1000M                         # Limite la mémoire consommée à 1000 Mo (1 Go), utile pour éviter les débordements.
    volumes:
      - mealie-data:/app/data/                  # Volume persistant pour conserver les recettes, les préférences, et la base de données.
    environment:

      # ======= Paramètres de base =======
      ALLOW_SIGNUP: true                        # Autorise la création de comptes utilisateurs (à désactiver en prod si LDAP est activé).
      PUID: 1000                                # UID (User ID) utilisé dans le container, pour l'utilisateur local.
      PGID: 1000                                # GID (Group ID) utilisé, généralement associé à l'utilisateur principal du système.
      TZ: Europe/Paris                          # Fuseau horaire (important pour les logs, planifications, etc.).
      MAX_WORKERS: 1                            # Nombre maximal de workers backend. Ici, 1 pour limiter la consommation.
      WEB_CONCURRENCY: 1                        # Nombre de processus web front-end (à 1 également pour limiter les ressources).
      BASE_URL: https://mealie.domaine.com      # L'URL publique utilisée pour les redirections et les liens dans l'interface.

      # ======= Configuration LDAP (authentification externe) =======
      LDAP_AUTH_ENABLED: true                   # Active l'authentification LDAP pour les utilisateurs (au lieu de comptes locaux).
      LDAP_SERVER_URL: ldap://<ip>:389  # Adresse IP et port du serveur LDAP (ici sans TLS).
      LDAP_TLS_INSECURE: true                   # Ignore les vérifications de certificat TLS (utile en local, à éviter en prod).
      LDAP_ENABLE_STARTTLS: false               # Désactive StartTLS (utilisé pour sécuriser une connexion LDAP simple).
      LDAP_BASE_DN: ou=people,dc=domaine,dc=com  # DN de base à partir duquel les utilisateurs sont recherchés.
      LDAP_QUERY_BIND: cn=admin,dc=domaine,dc=com   # Compte utilisé pour interroger
```

```
l’annuaire LDAP.
      LDAP_QUERY_PASSWORD: motdepasse           # Mot de passe du compte de requête LDAP (*à
stocker dans un secret en prod*).
      LDAP_USER_FILTER: (memberOf=cn=mealie,ou=groups,dc=domaine,dc=com)
        # Filtre pour restreindre l'accès aux utilisateurs membres du groupe "mealie".
      LDAP_ADMIN_FILTER: (memberOf=cn=mealie-admin,ou=groups,dc=domaine,dc=com)
        # Filtre pour reconnaître les administrateurs du service, membres du groupe "mealie-
admin".
      LDAP_ID_ATTRIBUTE: uid                    # Attribut LDAP utilisé comme identifiant
unique.
      LDAP_NAME_ATTRIBUTE: cn                   # Attribut utilisé comme nom affiché.
      LDAP_MAIL_ATTRIBUTE: mail                 # Attribut utilisé pour l’adresse e-mail.
      LOG_LEVEL: debug                          # Niveau de log (debug = très verbeux, utile
en phase de test).

volumes:
  mealie-data:                                  # Volume Docker nommé, monté dans /app/data
pour la persistance.
```

Dans mon cas, j'utilise un serveur LDAP pour l'authentification, plus simple, au vu du nombre des différents services que j'héberge.
Mealie est cependant capable d'une gestion utilisateurs en interne, ainsi qu'une gestion de "Foyer", similaire à des groupes.

Démarrage

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

```
docker compose up -d
```

Revision #1
Created 2025-10-29 12:44:31 UTC
Updated 2025-10-29 12:44:31 UTC