

# Vaultwarden passwordmanager

- [Installation](#)
- [Backup & Restore](#)

# Installation

docker-compose.yml

```
version: '2'
services:
  vaultwarden:
    image: ghcr.io/dani-garcia/vaultwarden
    container_name: vaultwarden
    restart: unless-stopped
    volumes:
      - ./data:/data/
    environment:
      WEBSOCKET_ENABLED: "true" # Enable WebSocket notifications.
      ADMIN_TOKEN: ${VAULTWARDEN_ADMIN_TOKEN}
      PASSWORD_ITERATIONS: 500000
    labels:
      traefik.enable: true
      traefik.http.services.vaultwarden-service.loadbalancer.server.port: 80
      traefik.http.services.vaultwarden-ws-service.loadbalancer.server.port: 3012

      traefik.http.routers.vaultwarden-admin.entrypoints: websecure
      traefik.http.routers.vaultwarden-admin.middlewares: secured@file
      traefik.http.routers.vaultwarden-admin.rule: Host(`vaultwarden.${SITE}`) && PathPrefix(`/admin`)
      traefik.http.routers.vaultwarden-admin.service: vaultwarden-service

      traefik.http.routers.vaultwarden-user.entrypoints: websecure
      traefik.http.routers.vaultwarden-user.rule: Host(`vaultwarden.${SITE}`) && !PathPrefix(`/admin`)
      traefik.http.routers.vaultwarden-user.service: vaultwarden-service

      traefik.http.routers.vaultwarden-ws.entrypoints: websecure
      traefik.http.routers.vaultwarden-ws.rule: Host(`vaultwarden.${SITE}`) && Path(`/notifications/hub`)
      traefik.http.routers.vaultwarden-ws.service: vaultwarden-ws-service
```

## Getting the admin token

See [here](#).

1. Run the following command and choose a password.

```
docker exec -it vwcontainer /vaultwarden hash
```

2. Although the documentation states that it is not necessary, replace all `$` with `$$` in the string.
3. Put the string into the `.env` file. For example:

```
VAULTWARDEN_ADMIN_TOKEN=$$argon2id$$v=19$$m=19456,t=2,p=1$$UUZxK1FZMkZoRHFQRIVr  
TXZvS0E3bHpNQW55c2dBN2NORzdsa0Nxd1JhND0$$cUold+JBUsJutlG4rfDZayExfjq4TCt48aBc9qsc3UI
```

- You can test if websockets are working by opening vaultwarden in two different browsers/devices and creating an entry in one of them. It should automatically sync over to the other.

## See also

- [https://github.com/dani-garcia/vaultwarden/wiki/Enabling-admin-page#secure-the-admin\\_token](https://github.com/dani-garcia/vaultwarden/wiki/Enabling-admin-page#secure-the-admin_token)
- <https://github.com/dani-garcia/vaultwarden/wiki/Proxy-examples#traefik-v1-labels-migrated-to-traefik-v2>

# Backup & Restore

For the backup we can use the `vaultwarden_backup` image.

`docker-compose.yml`

```
services:
  vaultwarden_backup:
    image: jmqm/vaultwarden_backup:latest
    container_name: vaultwarden_backup
    network_mode: none
    # command: manual
    volumes:
      - ./data:/data:ro # Read-only
      - ./backup:/backups
      - /etc/localtime:/etc/localtime:ro # Container uses date from host.
    environment:
      - DELETE_AFTER=30
      - CRON_TIME=* */24 * * * # Runs at 12:00 AM.
      - UID=${PUID}
      - GID=${PGID}
```

## Backing up

Simply set the `CRON_TIME` to the desired backup frequency. If you want to force a backup, uncomment `command: manual` and restart the container.

## Restore

Create a new folder (e.g. `mkdir ~/vaultwarden_restore`). Unzip the desired tarball with `tar xvf <backup>.tar.xz -C ~/vaultwarden_restore`. Then, simply move over all extracted files and folders to the `data` folder of a new install.

## Things to remember

- Make sure that permission are set correctly and check backups periodically

## See also

- [https://github.com/jmqm/vaultwarden\\_backup](https://github.com/jmqm/vaultwarden_backup)