

Installation

- Docker installation

Docker installation

Beschreibung:

Vaultwarden Docker installation.

Docker installieren

```
apt install docker.io docker-compose curl
```

Verzeichnisse erstellen

```
mkdir -p /root/vaultwarden/nginx/ssl  
mkdir -p /root/vaultwarden/vw-data
```

Vaultwarden braucht nur eine Docker compose datei.
Dort die Domain rein.

```
nano /root/vaultwarden/docker-compose.yml
```

```
services:  
  vaultwarden:  
    image: vaultwarden/server:latest  
    container_name: vaultwarden  
    restart: unless-stopped  
    environment:  
      DOMAIN: ${DOMAIN}  
      SIGNUPS_ALLOWED: ${SIGNUPS_ALLOWED}  
      ADMIN_TOKEN: ${ADMIN_TOKEN}  
    volumes:  
      - ./vw-data:/data/  
    ports:  
      - 80:80
```

Selbstsigniertes Zertifikat mit nginx:

Möchten wir das ganze mit einem selbstsigniertem ssl Zertifikat haben aber lokal dann noch ein nginx dazu.

Dann müssen wir beim vaultwarden port 80 wieder löschen

```
nginx:
  image: nginx:stable
  container_name: nginx-proxy
  volumes:
    - ./nginx-proxy.conf:/etc/nginx/nginx.conf:ro
    - ./nginx/ssl:/etc/nginx/ssl
  ports:
    - "80:80"
    - "443:443"
  depends_on:
    - vaultwarden
  restart: always
```

Ein script

```
nano /root/vaultwarden/generate_certs.sh
```

```
#!/bin/bash
CERT_DIR="/root/vaultwarden/nginx/ssl"
mkdir -p $CERT_DIR

openssl req -x509 -nodes -days 36500 -newkey rsa:2048 -keyout $CERT_DIR/key.pem -out $CERT_DIR/cert.pem -
subj "/C=US/ST=Example/L=Example/O=Example/CN=example.com"
```

Script ausführbar machen

```
chmod +x /root/vaultwarden/generate_certs.sh
```

Script starten

```
cd /root/vaultwarden
./generate_certs.sh
```

Nginx config erstellen

```
nano /root/vaultwarden/nginx-proxy.conf
```

Inhalt

```
events {
  worker_connections 1024;
}

http {
  server {
    listen 80;
    return 301 https://$host$request_uri;
  }

  server {
    listen 443 ssl;
    ssl_certificate /etc/nginx/ssl/cert.pem;
    ssl_certificate_key /etc/nginx/ssl/key.pem;

    location / {
      proxy_pass http://vaultwarden:80;
      proxy_set_header Host $host;
      proxy_set_header X-Real-IP $remote_addr;
      proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
      proxy_set_header X-Forwarded-Proto $scheme;
      proxy_set_header Upgrade websocket;
      proxy_set_header Connection Upgrade;
      proxy_read_timeout 20d;
      client_max_body_size 100M;
    }
  }
}
```

Mit Lets Encrypt Zertifikat

Dazu in der docker-compose.yml die Ports 80 bei vaultwarden raus und diesen Abschnitt hinzufügen

```
caddy:
  image: caddy:latest
  container_name: vaultwarden_caddy
  ports:
```

```
- "80:80"
- "443:443"
restart: always
environment:
  - CADDY_DOMAIN=${DOMAIN}
volumes:
  - ./caddy_data:/data
  - ./caddy_config:/config
command: caddy reverse-proxy --from ${DOMAIN} --to vaultwarden:80
```

env datei erstellen/abändern

```
nano /root/vaultwarden/.env
```

Inhalt

```
DOMAIN=vault.example.com
```

Container starten:

nun mit

```
docker-compose up -d
```

starten

Konto erstellen:

Nun über Konto erstellen, konto Erstellen

Konto erstellen deaktivieren:

Möchte man nicht das sich neue Benutzer anmelden können,



Anmelden

E-Mail-Adresse (erforderlich)

⊗ Eingabe ist erforderlich.

E-Mail-Adresse merken

Fortsetzen

[Neu hier? Konto erstellen](#)

Dann die .env editieren bzw anlegen

```
nano /root/vaultwarden/.env
```

Und abändern/hinzufügen zu

```
SIGNUPS_ALLOWED=false
```

Dann mit

```
docker-compose down
```

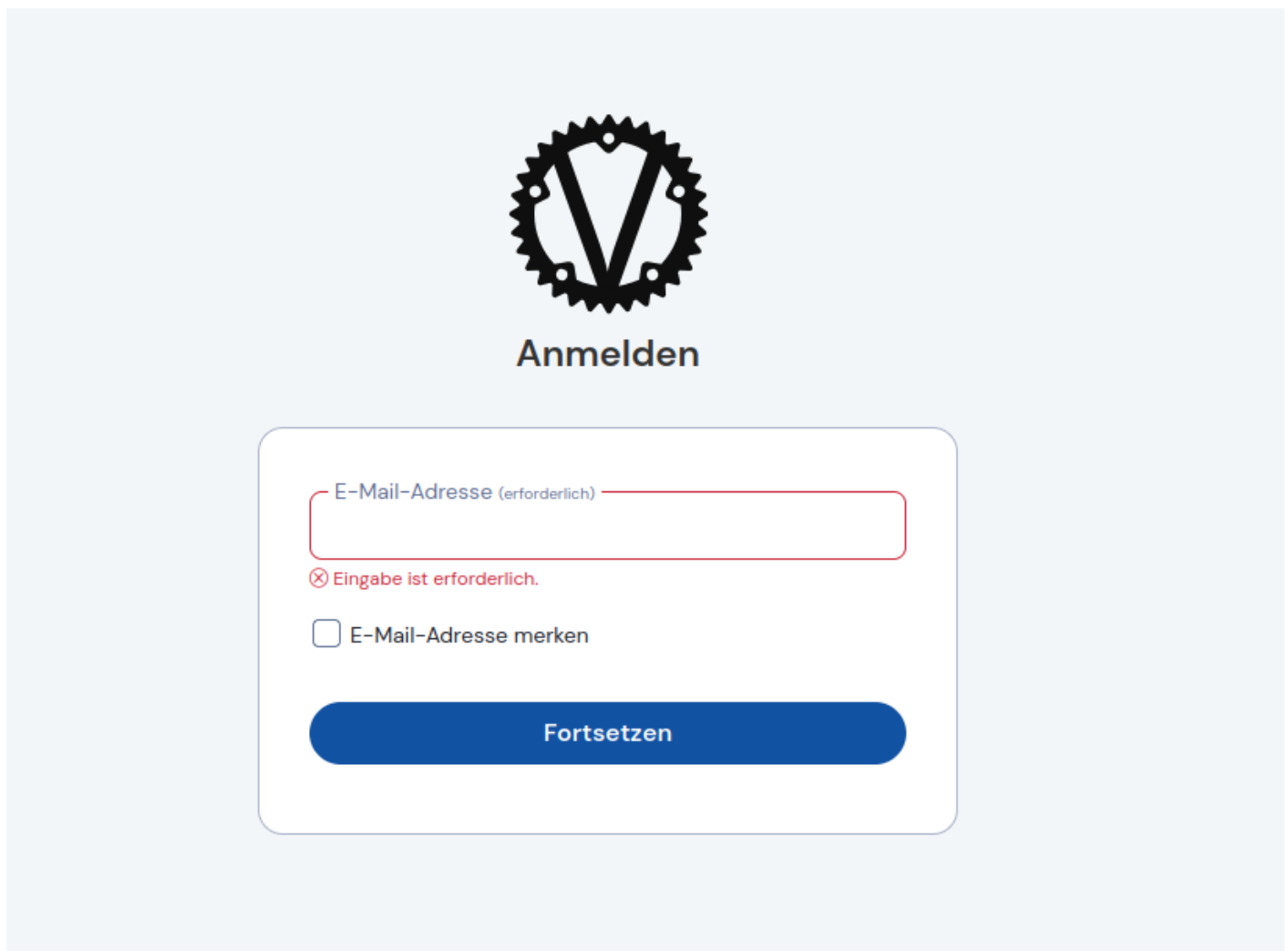
stoppen


und mit

```
docker-compose up -d
```

wieder starten

Nun kann man sich nicht mehr anmelden




Anmelden

E-Mail-Adresse (erforderlich)

⊗ Eingabe ist erforderlich.

E-Mail-Adresse merken

Fortsetzen

Sollte das Anmelden trotzdem da sein mit STRG+F5 Seite mit cache neuladen aufrufen

Admin Panel freischalten:

Einrichten:

Das Panel ist unter zu erreichen : <https://ip/hostname/admin>

Sollte dieser Fehler kommen

The admin panel is disabled, please configure the 'ADMIN_TOKEN' variable to enable it

in der .env bei diesem Eintrag ein Token einfügen, unbedingt neuen generieren, dieser ist ein Beispiel

```
ADMIN_TOKEN=$argon2id$v=19$m=65540,t=3,p=4$eHdqCtHRFFacXVXV0ltMIZKWGw5ZGliUHV2QjdVdVB1NUMwbVJ6SU5NZz0$LGVY+2LrZit7cS7MoEPWyHKUHwfhUq0Q2GQE9DFcGEg
```

Token generieren ein passwort vergeben was dann zu einem salt wird

Dazu wird das Paket argon2 benötigt

```
apt install argon2
```

```
echo -n "MySecretPassword" | argon2 "$(openssl rand -base64 32)" -e -id -k 65540 -t 3 -p 4
```

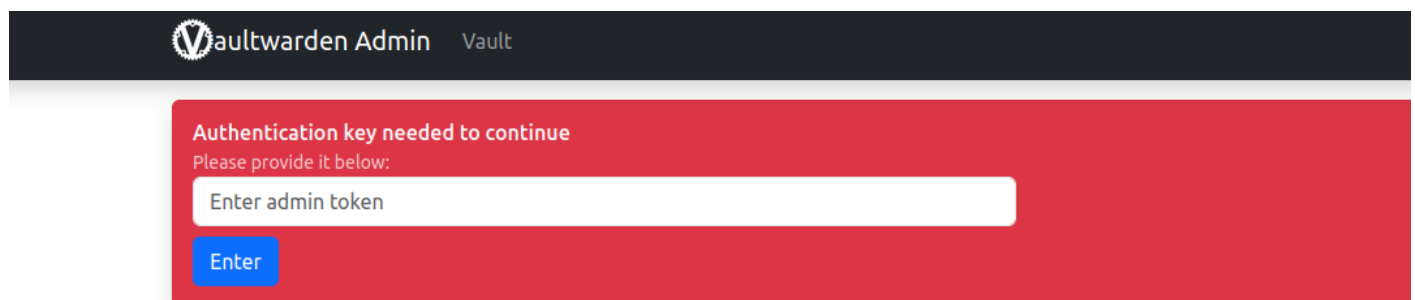
Ausgabe:

```
$argon2id$v=19$m=65540,t=3,p=4$eHdqCtHRFFacXVXV0ltMIZKWGw5ZGliUHV2QjdVdVB1NUMwbVJ6SU5NZz0$LGVY+2LrZit7cS7MoEPWyHKUHwfhUq0Q2GQE9DFcGEg
```

Contaeiner neustarten mit down und up -d

Anmelden:

Das Panel ist unter zu erreichen : <https://ip/hostname/admin>



Vaultwarden Admin Vault

Authentication key needed to continue
Please provide it below:

Enter

Nun den token eingeben, das ist das Passwort was ihr für MyPassword verwendet habt

Configuration

NOTE: The settings here override the environment variables. Once saved, it's recommended to stop setting them to avoid confusion. This does not apply to the read-only section, which can only be set via environment variables. Settings which are overridden are shown with a yellow colored background.

General settings

Advanced settings

Yubikey settings

Global Duo settings (Note that users can override them)

SMTP Email Settings

Email 2FA Settings

Read-Only Config

Backup Database

Save

Reset defaults

Einrichtung sends:

Damit Sends funktioniert muss der Haken drin sein und die korrekte IP-Adresse / Hostname angeben sein.

lässt sich einstellen im Admin Panel

Configuration

NOTE: The settings here override the environment variables. Once saved, it's recommended to stop setting them to avoid confusion. This does not apply to the read-only section, which can only be set via environment variables. Settings which are overridden are shown with a yellow colored background.

General settings

Domain URL Allow Sends Default: trueHIBP Api Key

Show/hide