

Immich Stack mit Crowdsec und Traefik

- [Crowdsec-Einrichtung](#)

Crowdsec-Einrichtung

Folgendes Docker-Compose-File wurde zur Einrichtung des Crowdsec-Containers genutzt:

```
version: "3.8"

services:
  crowdsec:
    image: crowdsecurity/crowdsec:latest
    container_name: crowdsec
    restart: unless-stopped
    networks:
      traefik_proxy:
        ipv4_address: 192.168.101.3
    environment:
      COLLECTIONS: >
        crowdsecurity/traefik
        crowdsecurity/appsec-virtual-patching
        crowdsecurity/appsec-generic-rules
      GID: "${GID-1000}"
      TZ: Europe/Berlin
    volumes:
      - /opt/crowdsec/config:/etc/crowdsec
      - /opt/crowdsec/data:/var/lib/crowdsec/data
      - /opt/crowdsec/logs:/logs:ro
      - /opt/traefik/logs:/var/log/traefik:ro
    ports:
      - 8080:8080 # API für Bouncer/Plugin
      - 7422:7422 # API für AppSec

networks:

  traefik_proxy:
    external: true
```

Der AppSec-Listener muss während der Einrichtung gesondert aktiviert und konfiguriert werden. Hierzu wird die Datei `/opt/crowdsec/config/acquis.d/appsec.yaml` erstellt.

```
appsec_config: crowdsecurity/appsec-default
labels:
  type: appsec
listen_addr: 0.0.0.0:7422
source: appsec
```

Sollte ein zusätzliches oder anderes AppSec-Regelwerk genutzt werden, so kann dieses unter `appsec_config` konfiguriert werden. In diesem Beispiel wird das Standard-Regelwerk genutzt.

Einrichtung Crowdsec-Bouncer in Traefik

Das folgende Docker-Compose-File wurde für Traefik genutzt:

```
version: '3.8'

services:
  traefik:
    image: traefik:latest
    container_name: traefik
    restart: always
    command:
      - --experimental.plugins.crowdsec-bouncer.moduleName=github.com/maxlerebourg/crowdsec-
bouncer-traefik-plugin
      - --experimental.plugins.crowdsec-bouncer.version=v1.4.2
      - --api.dashboard=true
      - --api.insecure=false
      - --entrypoints.web.address=:80
      - --entrypoints.websecure.address=:443
      - --providers.docker=true
      - --providers.docker.exposedbydefault=false
      - --providers.file.filename=/etc/traefik/dynamic.yml
      - --providers.file.watch=true
      - --log.level=DEBUG
      - --log.filePath=/var/log/traefik/traefik.log
      - --accesslog=true
      - --accesslog.filePath=/var/log/traefik/access.log
      - --certificatesresolvers.le.acme.httpchallenge.entrypoint=web
```

- --certificatesresolvers.le.acme.email=service@petersen-it-services.de
- --certificatesresolvers.le.acme.storage=/letsencrypt/acme.json
- --entrypoints.web.http.redirections.entryPoint.to=websecure
- --entrypoints.web.http.redirections.entryPoint.scheme=https
- --entrypoints.web.http.redirections.entryPoint.permanent=true

ports:

- "80:80"
- "443:443"

networks:

traefik_proxy:

ipv4_address: 192.168.101.2

volumes:

- /var/run/docker.sock:/var/run/docker.sock:ro
- /opt/letsencrypt:/letsencrypt
- /opt/traefik/dynamic.yml:/etc/traefik/dynamic.yml:ro
- /opt/traefik/logs:/var/log/traefik

environment:

- TZ=Europe/Berlin

labels:

- "traefik.enable=true"
- "traefik.http.routers.traefik.rule=Host(`node01.immich.tpnxt.hostsysteme.de`)"
- "traefik.http.routers.traefik.entrypoints=websecure"
- "traefik.http.routers.traefik.service=api@internal"
- "traefik.http.routers.traefik.tls.certresolver=le"
- "traefik.http.routers.traefik.middlewares=crowdsec@file"

networks:

traefik_proxy:

external: true

volumes:

traefik_letsencrypt:

Der Traefik-Container nutzt das externe Plug-in `crowdsec-bouncer-traefik-plugin`, definiert in den Zeilen 9 bis 10. Des Weiteren wird in Zeile 50 die Crowdsec-Middleware definiert. Konfiguriert wird die Middleware über das dedizierte Config-File `/opt/traefik/dynamic.yml`. Eingestellt wird unter anderem die Verbindung zur lokalen Crowdsec-API, der Caching-Mode (`crowdsecMode`), das Verhalten, falls der Appsec-Endpoint des Crowdsec-Containers nicht erreichbar ist, sowie eine

Whitelist erlaubter Reverse-Proxies.

```
http:
  middlewares:

    crowdsec:
      plugin:
        crowdsec-bouncer:
          enabled: true
          defaultDecisionSeconds: 60
          crowdsecMode: stream
          crowdsecAppsecEnabled: true
          crowdsecAppsecHost: 192.168.101.3:7422
          crowdsecAppsecFailureBlock: true
          crowdsecAppsecUnreachableBlock: true
          crowdsecLapiKey: <REDACTED>
          crowdsecLapiHost: 192.168.101.3:8080
          crowdsecLapiScheme: http
          crowdsecLapiTLSInsecureVerify: false
          TrustUpstream: true
          forwardedHeadersTrustedIPs:
            - 192.168.101.0/24
            - 172.0.0.0/8
```

Nutzung der Crowdsec-Middleware durch Immich

Folgendes Docker-Compose-File wurde für Immich genutzt:

```
#
# WARNING: To install Immich, follow our guide: https://immich.app/docs/install/docker-compose
#
# Make sure to use the docker-compose.yml of the current release:
#
# https://github.com/immich-app/immich/releases/latest/download/docker-compose.yml
#
# The compose file on main may not be compatible with the latest release.

name: immich
```

services:

immich-server:

container_name: immich_server-9999

image: ghcr.io/immich-app/immich-server:release

extends:

file: hwaccel.transcoding.yml

service: cpu # set to one of [nvenc, quicksync, rkmpp, vaapi, vaapi-wsl] for

accelerated transcoding

volumes:

Do not edit the next line. If you want to change the media storage location on your system, edit the value of UPLOAD_LOCATION in the .env file

- /opt/immich/data/9999/:/usr/src/app/upload

- /etc/localtime:/etc/localtime:ro

networks:

- immich_net_9999

- traefik_proxy

environment:

DB_PASSWORD: Hallo123-

DB_USERNAME: immich-psql-9999

DB_DATABASE_NAME: immich-9999

depends_on:

- redis

- database

restart: always

healthcheck:

disable: false

labels:

- "traefik.enable=true"

- "traefik.docker.network=traefik_proxy"

- "traefik.http.routers.immich-9999.entrypoints=websecure"

- "traefik.http.routers.immich-9999.tls.certresolver=le"

- "traefik.http.services.immich-9999.loadbalancer.server.port=2283"

- "traefik.http.routers.immich-9999.rule=Host(`test.tpnxt.cloud`)"

- "traefik.http.routers.immich-9999.middlewares=crowdsec@file"

immich-machine-learning:

container_name: immich_machine_learning-9999

For hardware acceleration, add one of -[armnn, cuda, rocm, openvino, rknn] to the image

tag.

```
# Example tag: ${IMMICH_VERSION:-release}-cuda
```

```
image: ghcr.io/immich-app/immich-machine-learning:release
```

```
# extends: # uncomment this section for hardware acceleration - see
```

<https://immich.app/docs/features/ml-hardware-acceleration>

```
# file: hwaccel.ml.yml
```

```
# service: cpu # set to one of [armnn, cuda, rocm, openvino, openvino-wsl, rknn] for  
accelerated inference - use the `-wsl` version for WSL2 where applicable
```

```
volumes:
```

```
- model-cache:/cache
```

```
networks:
```

```
- immich_net_9999
```

```
restart: always
```

```
healthcheck:
```

```
disable: false
```

```
redis:
```

```
container_name: immich_redis-9999
```

```
image: docker.io/valkey/valkey:8-
```

bookworm@sha256:fec42f399876eb6faf9e008570597741c87ff7662a54185593e74b09ce83d177

```
networks:
```

```
- immich_net_9999
```

```
healthcheck:
```

```
test: redis-cli ping || exit 1
```

```
restart: always
```

```
database:
```

```
container_name: immich_postgres-9999
```

```
image: ghcr.io/immich-app/postgres:14-vectorchord0.4.3-pgvector0.2.0
```

```
environment:
```

```
POSTGRES_PASSWORD: Hallo123-
```

```
POSTGRES_USER: immich-psql-9999
```

```
POSTGRES_DB: immich-9999
```

```
POSTGRES_INITDB_ARGS: '--data-checksums'
```

```
# Uncomment the DB_STORAGE_TYPE: 'HDD' var if your database isn't stored on SSDs
```

```
# DB_STORAGE_TYPE: 'HDD'
```

```
volumes:
```

```
# Do not edit the next line. If you want to change the database storage location on your  
system, edit the value of DB_DATA_LOCATION in the .env file
```

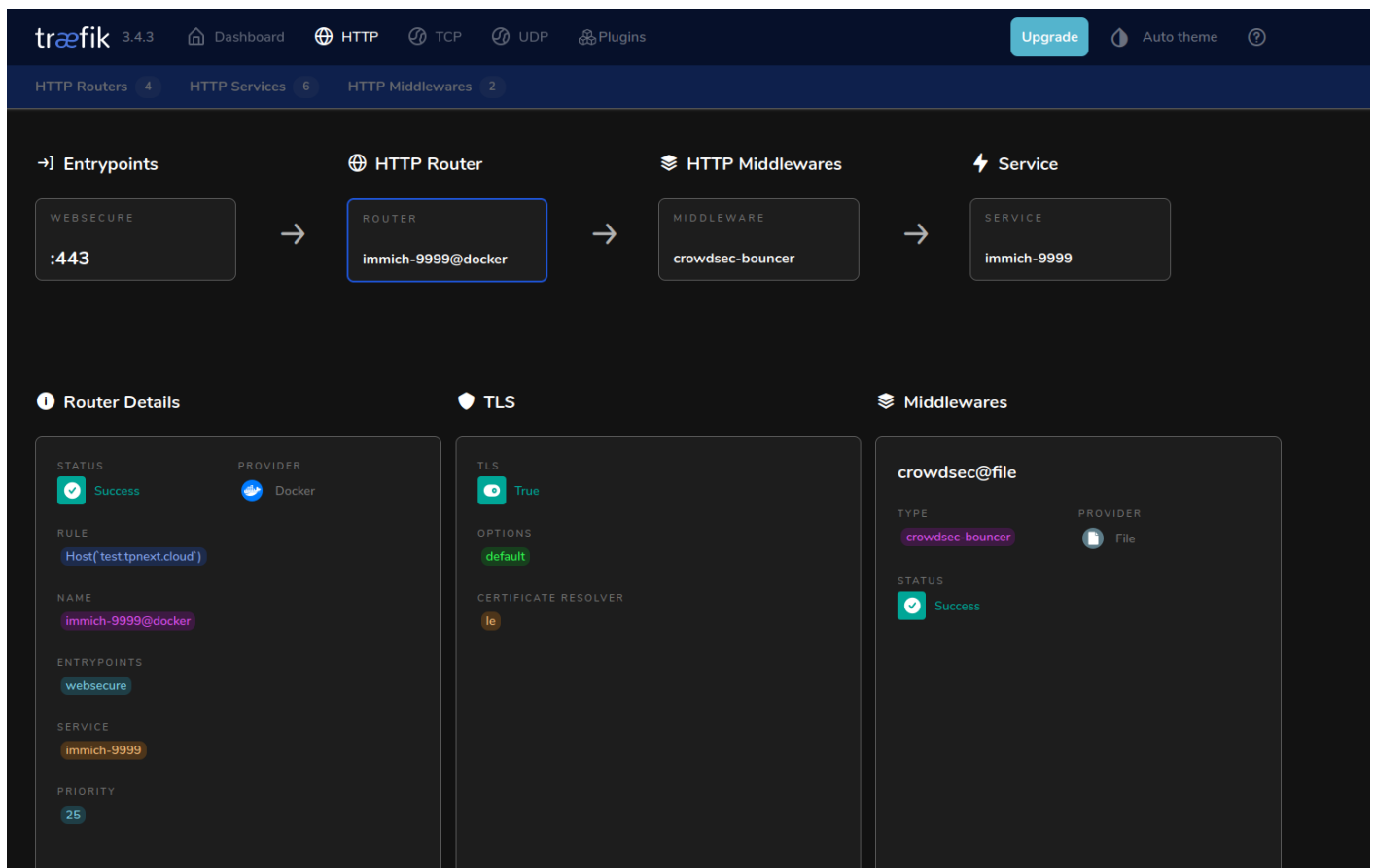
```
- /opt/immich/db/9999:/var/lib/postgresql/data
```

```
networks:
  - immich_net_9999
restart: always
```

```
volumes:
  model-cache:
```

```
networks:
  immich_net_9999:
    driver: bridge
  traefik_proxy:
    external: true
```

In Zeile 43 wird die Nutzung der Middleware `crowdsec@file` im Traefik-Router konfiguriert. Die Nutzung dieser Middleware kann im Traefik-Dashboard überprüft werden.



Auswertung der Traefik-Logs durch Crowdsec

In Zeilen 21 bis 22 der Docker-Traefik-Konfiguration wurde die Erstellung von Access-Logs unter dem Pfad `/var/log/traefik/access.log` konfiguriert. In Zeile 41 wird das Log-Verzeichnis `/var/log/traefik` persistent unter `/opt/traefik/logs` auf dem Docker-Host gespeichert.

Dieses persistente Verzeichnis wird in Zeile 22 der Docker-Crowdsec-Konfiguration als read-only unter `/var/log/traefik` innerhalb des Crowdsec-Containers gemountet.

Zu konsumierende Logs werden in `/opt/crowdsec/config/acquis.yaml` definiert.

```
poll_without_inotify: false
filenames:
  - /var/log/traefik/access.log
labels:
  type: traefik
```