Dynamische Konfiguration

Traefik bietet verschiedene Möglichkeiten der Konfiguration. Bisher wurden alle Konfigurationen über die Docker Compose Datei vorgenommen. Stattdessen kann auch eine dynamische Konfiguration in Form einer oder mehrerer YAML-Datei(en) verwendet werden.

Der Große Vorteil dieser YAML-Dateien liegt in der Möglichkeit, Traefik ohne Neustart anzupassen. Einfach die Datei(en) ändern und Traefik erkennt automatisch die Änderungen und lädt seine Konfiguration neu. Hierdurch gibt es keine Verbindungsabbrüche.

Zuerst erstellen wir einen Ordner samt Unterordner im Verzeichnis der Docker Compose Datei, z. B. mit den Namen *traefik_data/conf* und wechseln in dieses.

```
mkdir traefik_data
mkdir traefik_data/conf
cd traefik_data/conf
```

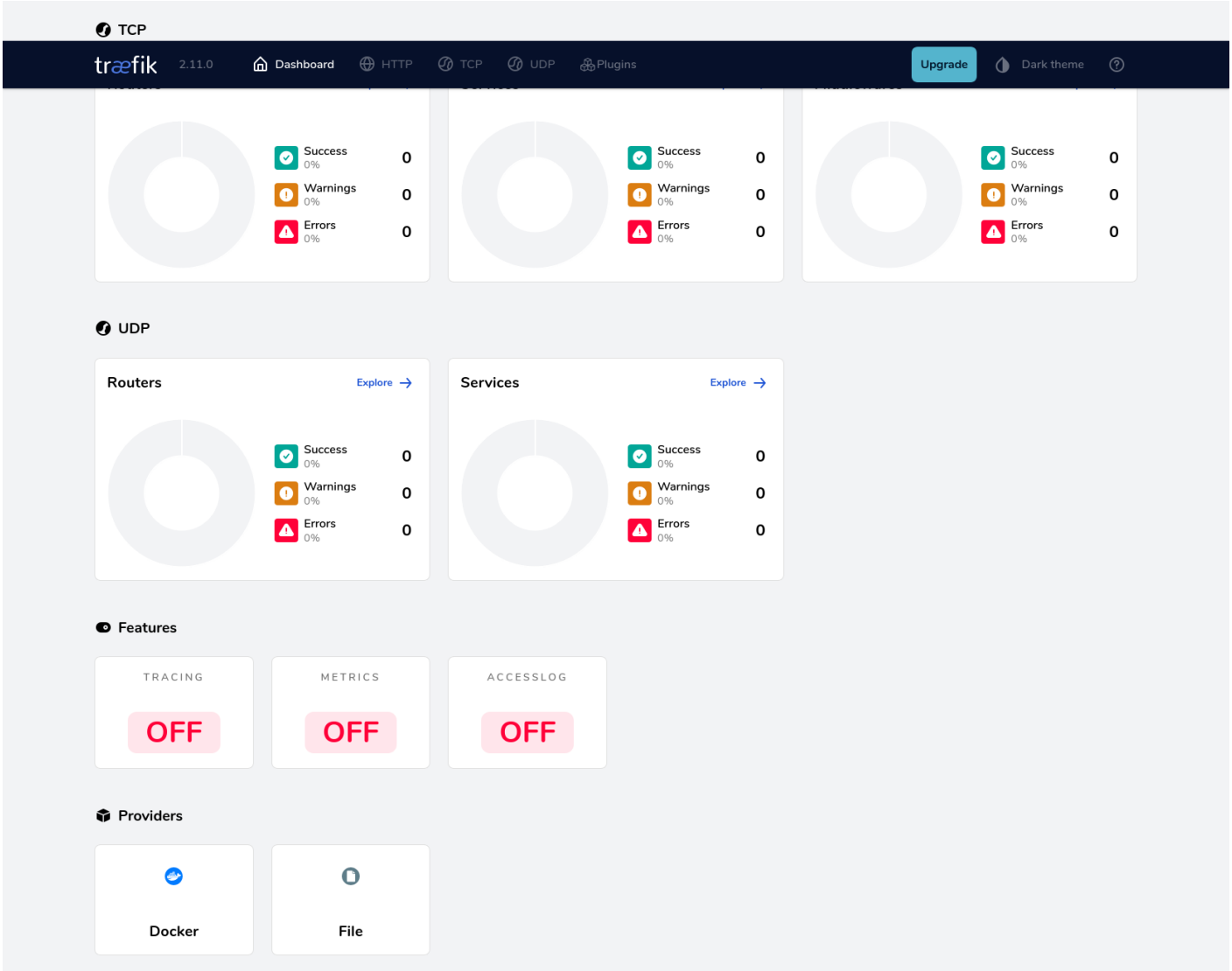
Als Grundlage nehmen wir die Compose Konfiguration von der Seite Installation und erweitern diese um ein Volume, in welchem wir später die dynamischen Konfigurationsdateien ablegen. Zusätzlich ist unter Command noch der Pfad zu dem Ordner mit den Konfigurationsdateien im Container anzugeben. Der soeben erstellte Ordner wird als Volume eingebunden.

```
services:
  traefik:
    image: "traefik"
    container_name: "traefik"
    command:
      - "--providers.file.directory=/traefik/conf"
      - "--api.insecure=true"
      - "--providers.docker=true"
      - "--providers.docker.exposedbydefault=false"
      - "--providers.docker.network=traefik"
      - "--entrypoints.web.address=:80"
    ports:
      - "80:80"
      - "8080:8080"
    volumes:
```

```
      - /var/run/docker.sock:/var/run/docker.sock:ro
      - /pfad/zu/traefik/traefik_data:/traefik
    networks:
      - traefik
networks:
  traefik:
    external: true
```

Um die Docker Compose Konfiguration auszuführen, kann am besten in das Verzeichnis der YAML Datei gewechselt werden. Danach wird je nach nach gewählter Installation `sudo docker-compose up -d` oder `sudo docker compose up -d` (keine Bindestrich zwischen docker und compose) eingegeben, um die Standard Konfiguration `docker-compose.yml` zu starten. Compose erstellt dann die gewünschten Container mit den angegeben Optionen. Sollten die Container bereits mit dieser Compose Konfiguration erstellt worden sein, so werden die Container in dieser neu erstellt, dessen Konfiguration geändert wurde.

Nachdem Traefik läuft, einmal die korrekte Ausführung prüfen, entweder durch den Aufruf von http://localhost:8080 (Dashboard) oder indem in die Docker Logs geschaut wird. Im Dashboard kann unten auf der Seite unter *Providers* geprüft werden, ob der Ordner für die Konfigurationsdateien überwacht wird. Dort sollte neben Docker nun auch File stehen.



Nun fügen wir einen Dienst über eine Konfigurationsdatei hinzu. Am besten nehmen wir hierzu das Beispiel von der Seite Dienst mit Basic Auth absichern.

Angenommen wir wollen mehrere Dienste mit Basic-Auth absichern, dann wäre es am besten, die Konfiguration der Benutzer in einer dynamischen Konfigurationsdatei vorzunehmen. Wir streichen also die Zeile traefik.http.middlewares.whoami-auth.basicauth.users aus der Compose Datei und ändern den Namen der Zeile für *middlewares* ab. Die neue Compose Datei sieht wie folgt aus.

```
services:
  whoami:
    image: "traefik/whoami"
    container_name: "simple-service"
    labels:
      - "traefik.enable=true"
      - "traefik.http.routers.whoami.rule=Host(`whoami.localhost`)"
      - "traefik.http.services.whoami.loadbalancer.server.port=80"
      - "traefik.http.routers.whoami.middlewares=basic-auth@file"
```

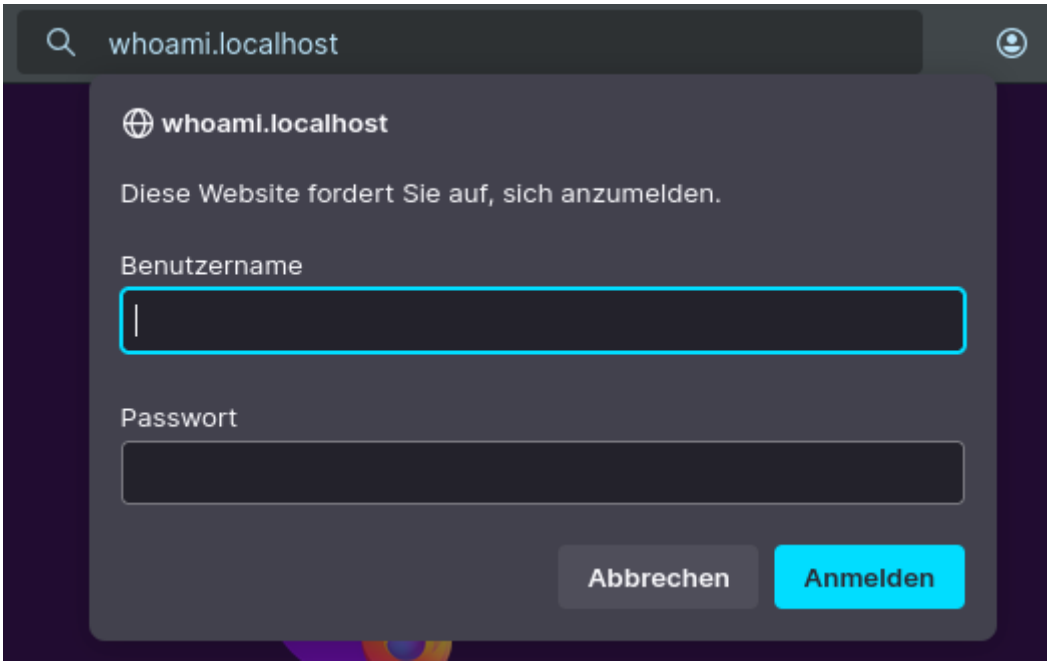
```
    networks:
      - traefik
networks:
  traefik:
    name: traefik
    external: true
```

Die Compose Konfiguration kann nun gestartet werden.

Anschließend erstellen wir die Datei `traefik_data/conf/basic-auth.yml` mit folgendem Inhalt.

```
http:
  middlewares:
    basic-auth:
      basicauth:
        users:
          benutzername:$2a$12$74NsF8MzGC25qO5tYoGfnO1Tg9I9c5Fcbu/zwi1/paDfdYD8eeUxy
```

Nun können wir direkt den neuen Dienst http://whoami.localhost aufrufen und es erscheint eine Login Abfrage.



Für die Generierung des Passworts kann folgende Seite verwendet werden: Bcrypt-Generator

Es können beliebige weitere Dateien erstellt werden um Plugins oder Seiten oder Dienst usw. zu konfigurieren. Die Einstellungen werden direkt angewendet, sobald die Datei gespeichert wird. Sollten Syntax-Fehler gefunden werden, bleibt die alte Konfiguration bestehen und im Docker Log werden Fehler angezeigt.

Version #1
Erstellt: 2024-03-21 18:28:37 CET von Marcel
Zuletzt aktualisiert: 2024-03-21 19:15:33 CET von Marcel