traefik

Docker Label Configuration

Base Labels

This is the minimum set of labels you need to expose a container to traefik:

```
labels:
  traefik.enable: true
  traefik.http.routers.<service_name>.entrypoints: <ep1>, <ep2>
  traefik.http.routers.<service_name>.rule: Host(`host1`, `host2`)
  traefik.http.routers.<service_name>.tls: true
  traefik.http.routers.<service_name>.tls.certresolver: <cert_resolver>
```

Middleware configuration

To configure a middleware for a particular service, add the following label:

```
traefik.http.routers.<service_name>.middlewares: middlware@provider
```

Accessing on a non-default port

If a container exposes multiple ports or a non-default port:

```
traefik.http.services.<service_name>.loadbalancer.server.port: <port_num>
```

Networking

To expose only containers on a certain network to traefik, you must specify the providers.docker.network option as so:

```
providers:
  docker:
    endpoint:
        exposedByDefault: false # Require label in docker-compose file for each container
```

```
        network: <net_name>
        watch: true
```

> If traefik itself is running in a docker container, you must place it on the same network as the containers you want to expose.

TLS

Basic TLS configuration that enables resolvers for both single-domain and wildcard Let's Encrypt certificates, as well as staging certificates:

```
# ========== TLS Configuration ==========
tls:
  # Disable TLS version 1.0 and 1.1
    options:
    default:
      minVersion: VersionTLS12
      sniStrict: true

  certificatesResolvers:
    staging:
      acme:
        email: "email@email.com"
        storage: /etc/traefik/certs/acme.json
        caServer: "https://acme-staging-v02.api.letsencrypt.org/directory"
        tlsChallenge: {}

    production:
      acme:
        email: "email@email.com"
        storage: /etc/traefik/certs/acme.json
        caServer: "https://acme-v02.api.letsencrypt.org/directory"
        tlsChallenge: {}
```

> Wildcard certificates can only be obtained with the DNS-01 challenge. Therefore a resolver that uses these must have dnsChallenge configured accordingly.

Tailscale

When running traefik in a docker container, ensure that it has access to the tailscale socket to be able to issue TLS certificates through tailscale

---

Revision #9
Created 2023-05-01 11:33:09 UTC
Updated 2024-11-05 15:58:58 UTC