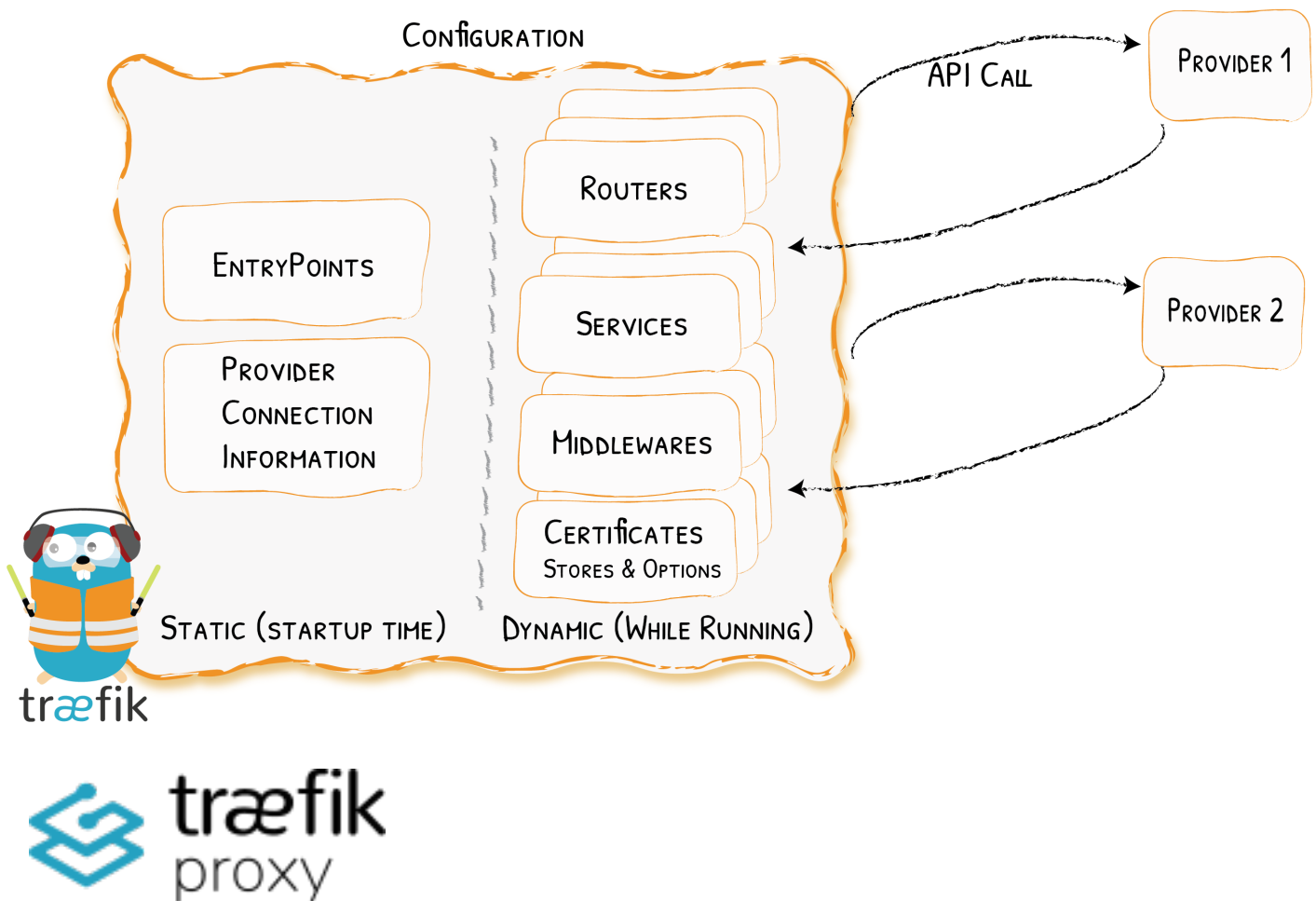


Configuration

- [Configuration Introduction](#)
- [Static Configuration File](#)
- [Dynamic Configuration File/s](#)

Configuration Introduction



Configuration in Traefik can refer to two different things:

- The fully dynamic routing configuration (referred to as the *dynamic configuration*)
- The startup configuration (referred to as the *static configuration*)

Elements in the *static configuration* set up connections to providers and define the entrypoints Traefik will listen to (these elements don't change often).

The *dynamic configuration* contains everything that defines how the requests are handled by your system. This configuration can change and is seamlessly hot-reloaded, without any request interruption or connection loss.

Incompatible Configuration

Please be aware that the old configurations for Traefik v1.x are NOT compatible with the v2.x config as of now. If you are running v2, please ensure you are using a v2 configuration. Old configurations from Traefik v2.x are NOT compatible.

The Static Configuration

There are three different, **mutually exclusive** (i.e. you can use only one at the same time), ways to define static configuration options in Traefik:

1. In a configuration file
2. In the command-line arguments
3. As environment variables

These ways are evaluated in the order listed above.

If no value was provided for a given option, a default value applies. Moreover, if an option has sub-options, and any of these sub-options is not specified, a default value will apply as well.

For example, the `--providers.docker` option is enough by itself to enable the docker provider, even though sub-options like `--providers.docker.endpoint` exist. Once positioned, this option sets (and resets) all the default values of the sub-options of `--providers.docker`.

Static files can be Configured in Format of TOML or YAML Format

See More about configuring the Static Config in this Wiki Post: [Static Configuration File](#)

See More about configuring the Static Config in this Wiki Post: [Static Configuration File](#)

The Dynamic Configuration

Traefik gets its *dynamic configuration* from providers: whether an orchestrator, a service registry, or a plain old configuration file.

Since this configuration is specific to your infrastructure choices, we invite you to refer to the dedicated section of this documentation.

Configuring Your Routers, Middlewares etc can be done in multiple ways. Either you can add Traefik Labels directly into your Application Docker Compose files Like this Format (- 'traefik.http.routers.bookstack.entrypoints=websecure') But they also can also be configured in Dynamic files. It's a Choice of what you prefer.

See More about configuring the Dynamic Config in this Wiki Post: [Dynamic Configuration Files](#)

Dynamic files can be Configured in Format of TOML or YAML Format

See More about configuring the Dynamic Config in this Wiki Post: [Dynamic Configuration Files](#)

HTTPS Certificates also belong to the dynamic configuration.

You can add / update / remove them without restarting your Traefik instance like mentioned in the beginning of this Post.

Static Configuration File



There are three different, mutually exclusive (i.e., you can use only one at the same time), ways to define static configuration options in Traefik:

- In a configuration file
- In the command-line arguments
- As environment variables

These ways are evaluated in the order listed above. If no value is provided for a given option, a default value applies. Moreover, if an option has sub-options and any of these sub-options are not specified, a default value will apply as well.

Static files can be Configured in Format of TOML or YAML Format.

I Prefer the YAML Format after working with TOML for 6 months because YAML is more intuitive for beginners.

How Traefik Loads the Static Configuration File

At startup, Traefik searches for static configuration in a file named `traefik.yml` (or `traefik.yaml` or `traefik.toml`) in:

- `/etc/traefik/`
- `$XDG_CONFIG_HOME/`
- `$HOME/.config/`
- `.` (the working directory).

Applying This Knowledge to Your Traefik Setup

Traefik will look for files named `traefik.yml`, `traefik.yaml`, or `traefik.toml` at the startup of the container. You need to mount the static configuration file into your Traefik container. Below is basic information on how to apply it in Docker Compose. For more details, refer to this post: [Bind Mounts](#).

I just call my static configuration file on my Linux host "`static.yaml`" so it is clear what is meant.

Examples of How to Apply

```
volumes:
  # Example 1
  - '/path/to/your/static.yml:/traefik.yml:ro'
  # Example 2
  - '/path/to/your/static.yaml:/traefik.yaml:ro'
  # Example 3
  - '/path/to/your/static.toml:/traefik.toml:ro'
```

Minimum Required Information Recommended in Your Static.yaml Example:

```
# Configuration for Traefik v3.3 by aeoneros

global:
  checkNewVersion: true
  sendAnonymousUsage: true

entryPoints:
  web:
    address: ":80"
  http:
```

```
redirections:
  entryPoint:
    to: websecure
    scheme: https
    permanent: true
```

```
websecure:
  address: ":443"
  http:
    tls:
      certResolver: "leresolver"
    domains:
      - main: 'yourdomain.com'
      sans:
        - '*.yourdomain.com'
```

```
tls:
  certificates:
    - stores: ["default"]
  stores:
    default:
      defaultCertificate: { }
```

```
log:
  filePath: "/traefik.log"
  level: "DEBUG"
```

```
accessLog:
  filePath: "/access.log"
  addInternals: true
  format: "json"
  bufferingSize: 200
  fields:
    defaultMode: "keep"
  headers:
    defaultMode: "keep"
  names:
    X-Forwarded-For: "keep"
```

```
api:
```

dashboard: false

insecure: false

providers:

docker:

exposedByDefault: false

network: "your_traefik_network"

swarm:

network: "your_traefik_network"

file:

watch: true

directory: "/dynamic/"

certificatesResolvers:

leresolver:

acme:

email: 'your-email@domain.com'

storage: '/acme.json'

caServer: 'https://acme-v02.api.letsencrypt.org/directory'

dnsChallenge:

provider: 'cloudflare'

delayBeforeCheck: '0'

resolvers:

- '1.1.1.1:53'

- '1.0.0.1:53'

- '8.8.8.8:53'

For more about dynamic configuration, see: [Dynamic Configuration Files](#).

Dynamic Configuration File/s



The dynamic configuration contains everything that defines how the

requests are handled by your system. This configuration can change and is seamlessly hot-reloaded, without any request interruption or connection loss.

How Traefik Loads the Dynamic Config File/s

Refer to the [Static Configuration Post](#) for context. Traefik loads the `static.yaml`, and within that file, the dynamic file(s) are referenced for loading.

To load your dynamic configurations, you can set them up in two ways. Either write all your dynamic content into a single file or split it into multiple files for better organization.

Example: Loading Configuration in One File

Needs to be set up in `static.yaml`. The `/dynamic.yaml` is the path inside the container and must be mounted using Docker Compose:

```
volumes:
  # Dynamic Configuration File
  - '/path/on/your/linuxhost/dynamic:/dynamic.yaml'
```

Example `static.yaml`:

```
providers:
  docker:
    exposedByDefault: false
    network: "management_net"
  swarm:
    network: "management_net"
file:
```

```
watch: true  
file: "/dynamic.yaml"
```

Example: Loading Multiple Configuration Files

This setup loads every file in the specified directory with the YAML format. The `/dynamic/` path is the path inside the container and must be mounted using Docker Compose:

```
volumes:  
  # Dynamic Configuration Folder  
  - '/path/on/your/linuxhost/dynamic:/dynamic/'
```

Example `static.yaml`:

```
providers:  
  docker:  
    exposedByDefault: false  
    network: "management_net"  
  swarm:  
    network: "management_net"  
file:  
  watch: true  
  directory: "/dynamic/"
```

Splitting Dynamic Files

Create multiple paths in your Traefik data folder. Assuming your base Traefik data folder is `/mnt/gluster mount/data/traefik_data`, you can structure it like this:

```
mkdir -p /mnt/gluster mount/data/traefik_data/dynamic  
touch /mnt/gluster mount/data/traefik_data/dynamic/http.middlewares.yaml  
touch /mnt/gluster mount/data/traefik_data/dynamic/http.routers.yaml  
touch /mnt/gluster mount/data/traefik_data/dynamic/http.services.yaml  
touch /mnt/gluster mount/data/traefik_data/dynamic/tls.options.yaml
```

Then you can write configurations into these files. Below are examples for each:

Example: `http.middlewares.yaml`

```
http:
  middlewares:
    my-secure-headers:
      headers:
        sslRedirect: true
        stsSeconds: 31536000
        stsIncludeSubdomains: true
        stsPreload: true
```

Example: `http.routers.yaml`

```
http:
  routers:
    my-router:
      rule: "Host(`example.com`)"
      entryPoints:
        - websecure
      service: my-service
      tls: {}
```

Example: `http.services.yaml`

```
http:
  services:
    my-service:
      loadBalancer:
        servers:
          - url: "http://192.168.1.10:8080"
```

Example: `tls.options.yaml`

```
tls:
  options:
    default:
      minVersion: VersionTLS12
      cipherSuites:
        - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
        - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
```