

Traefik

Reverse proxy with service discovery

AMT 2019

Olivier Liechti

Microservices

- **In the second project, we have built a simple application that illustrates the notion of “micro-service”:**
 - We have two services that implement business logic and expose it via a REST API.
 - Every service uses its own, independent datastore (nothing would prevent us to use MySQL for one, mongodb for the other, etc.).
 - We use HTTP to interact with the services. In some applications, micro-services also use message-oriented middleware (e.g. RabbitMQ) to communicate in the backend (e.g. to multicast data updates).
 - We have a docker-compose.yml that describes the complete application. At this stage, you most likely have 4 docker services in configuration (2 Spring boot apps, 2 databases).

Front-end?



HAUTE ÉCOLE
D'INGÉNIERIE ET DE GESTION
DU CANTON DE VAUD
www.heig-vd.ch

- **How would we build a front-end UI in this setup?**
 - A popular architecture, called “Single Page Application” consists in using a Javascript framework (Vue.js, React.js, Angular, etc.).
 - Javascript code is written to make AJAX requests to the REST API that we have implemented. When responses arrive, the page components are updated.
 - A common way to integrate the front-end app in the docker environment is to use a static HTTP server image (e.g. apache httpd, nginx).



single page app
(javascript)

2

1

http://???.api/users

http://???.api/courses

http://???./

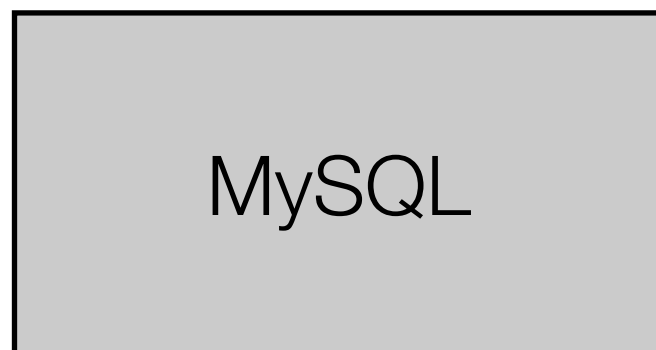
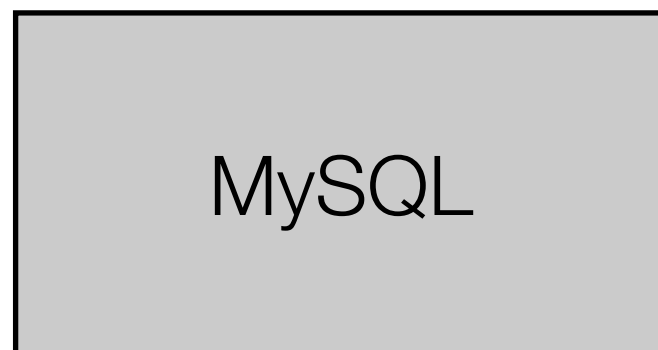
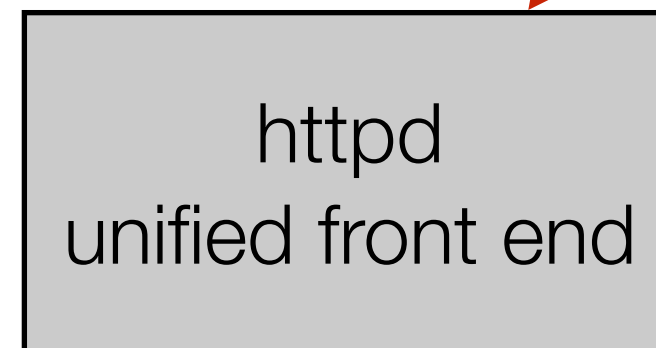
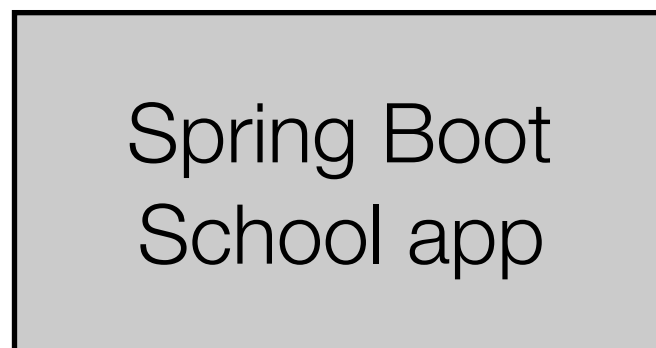
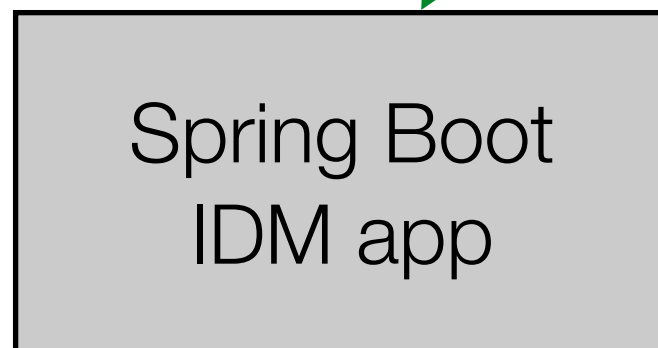
Spring Boot
IDM app

Spring Boot
School app

httpd
unified front end

MySQL

MySQL



Reverse proxy

- **Remember the RES lab, where we used a reverse proxy**
 - One goal was to deal with the single origin policy (without the setting up CORS): from the outside, there is a single logical website, serving static content (JS scripts) and dynamic content.
 - One goal was to create a single entry point in the system, and not to expose all services directly to the clients.
 - We used apache httpd (or nginx) and created a static configuration. In the configuration, we defined a mapping between URL prefixes and docker containers (ProxyPass, ProxyPassReverse directives).
 - With Traefik, we can do something very similar, but replace static configuration with dynamic configuration. The reverse proxy reacts to appearing and disappearing containers and adapts routing and load balancing.



single page app
(javascript)

2

1

<http://app.com/api/idm>
<http://app.com/api/courses>
<http://app.com/>

Reverse Proxy

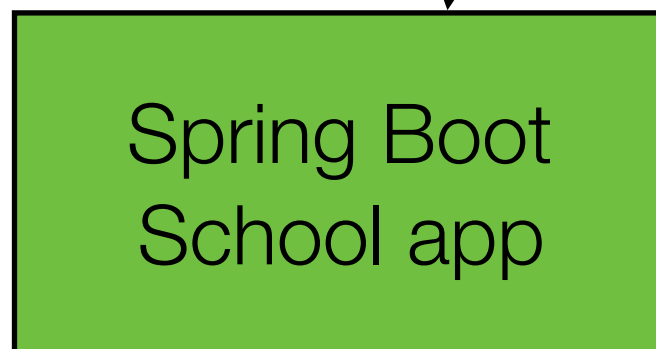
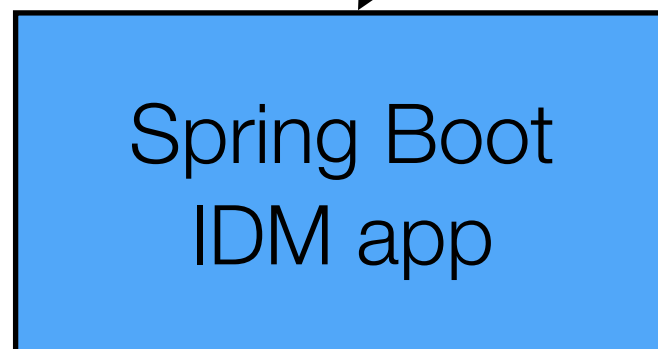
Spring Boot
IDM app

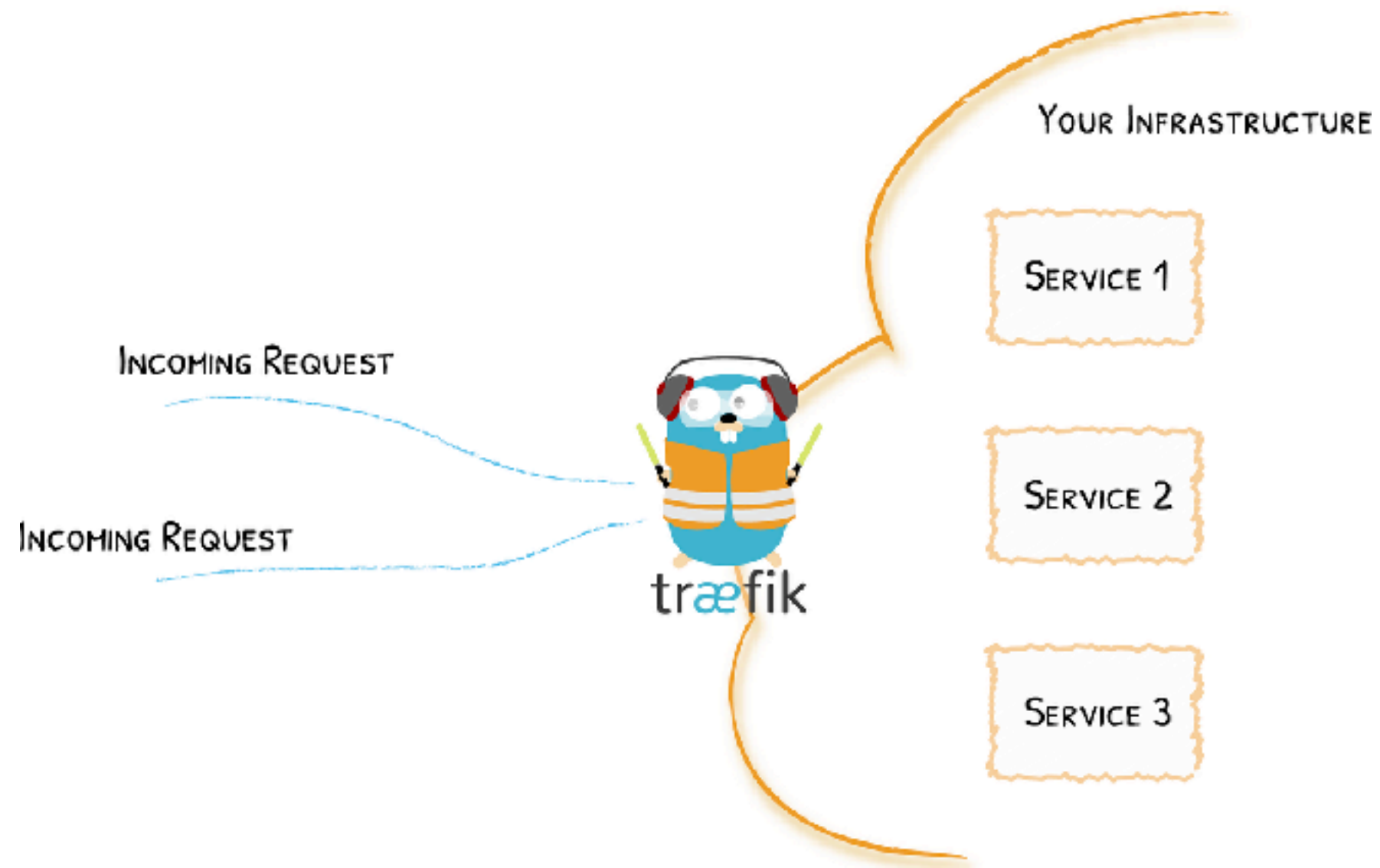
MySQL

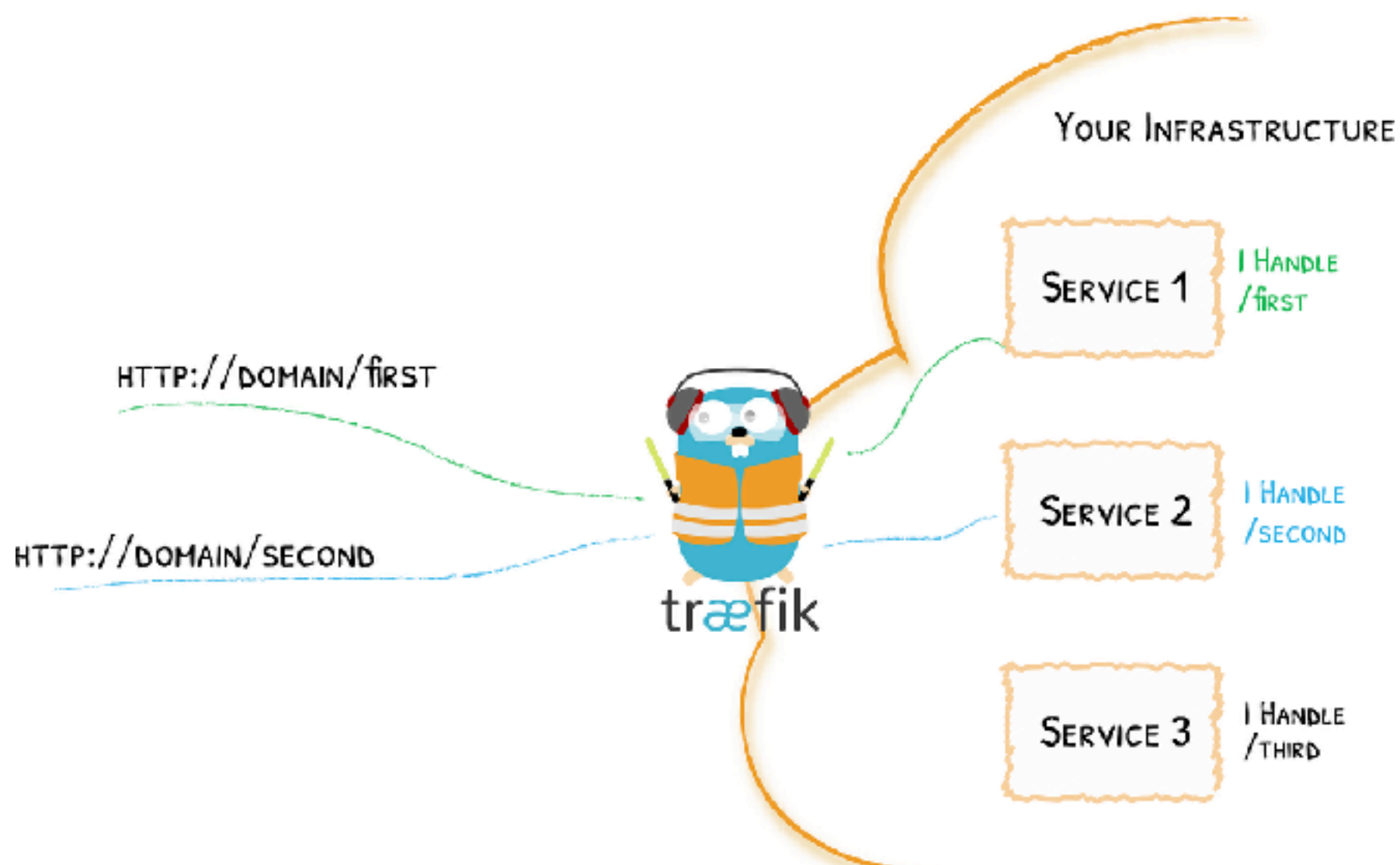
Spring Boot
School app

MySQL

httpd
unified front end

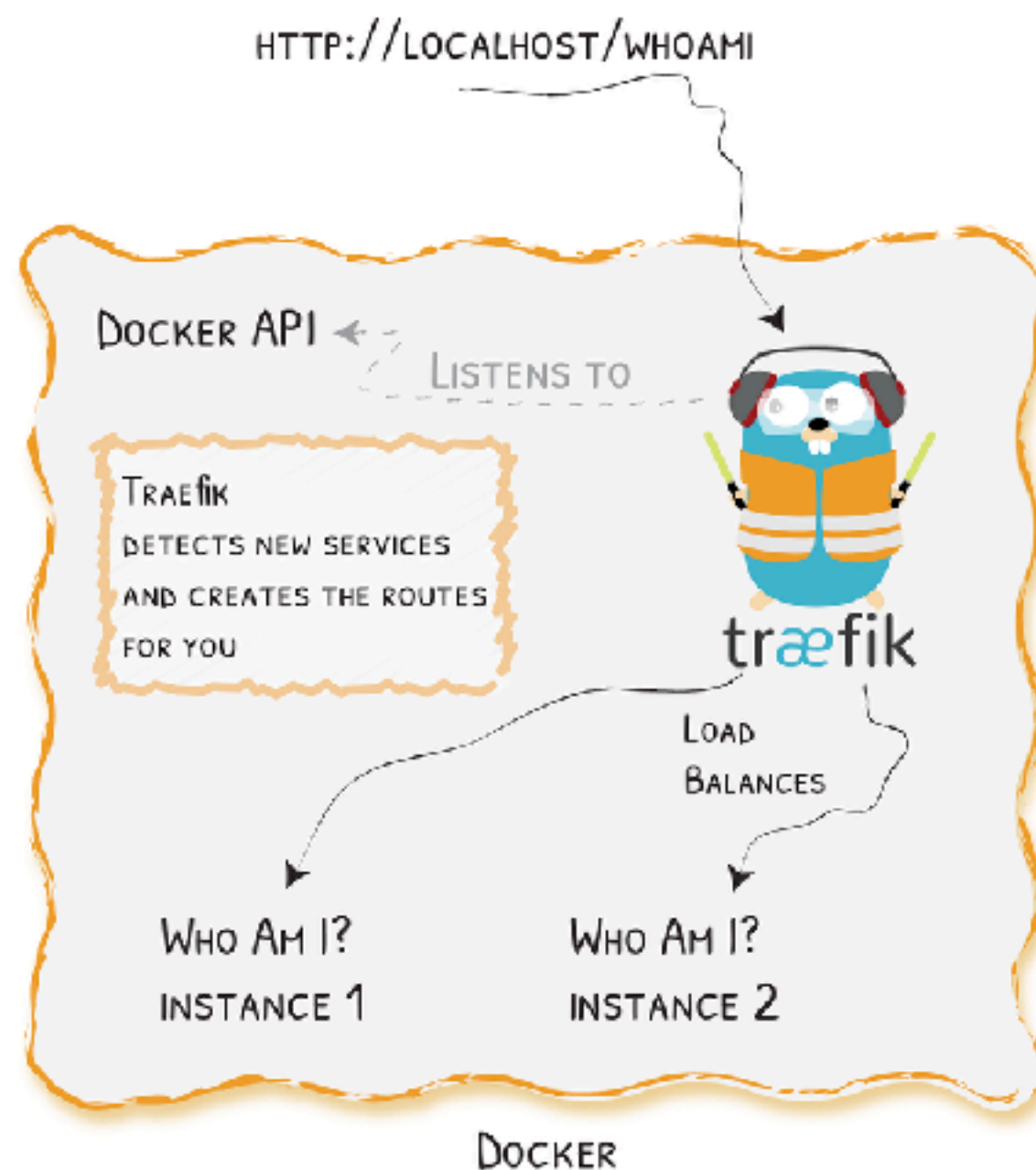






? How does Traefik discover the services?

Traefik is able to use your cluster API to discover the services and read the attached information. In Traefik, these connectors are called [providers](#) because they *provide* the configuration to Traefik. To learn more about them, read the [provider overview](#) section.

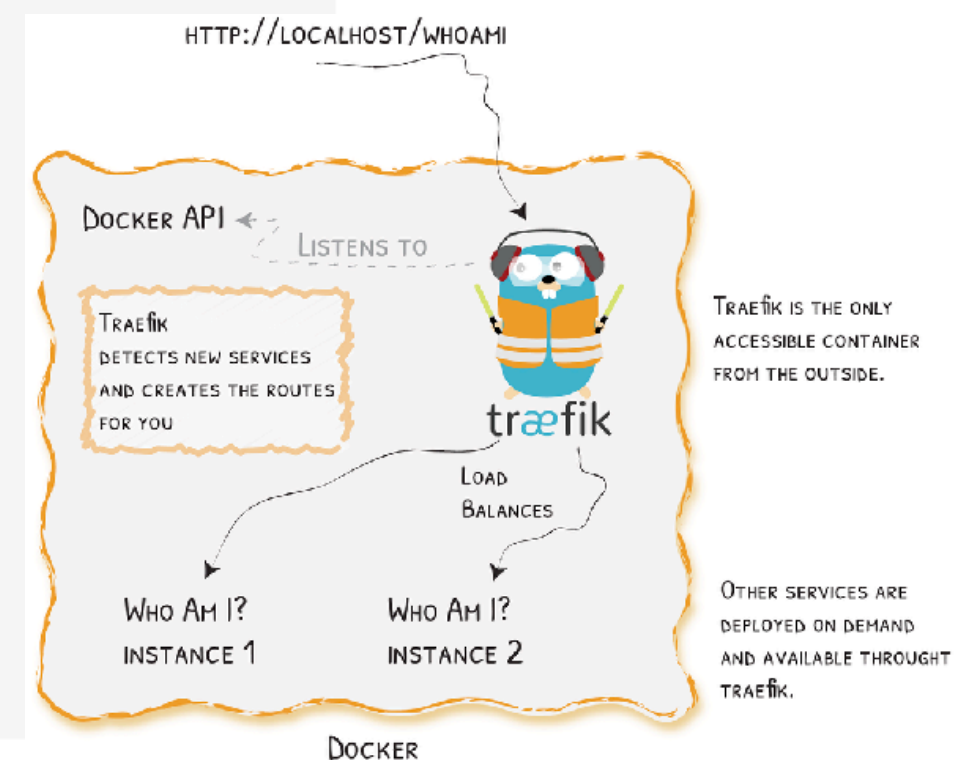


TRAEFIK IS THE ONLY
ACCESSIBLE CONTAINER
FROM THE OUTSIDE.

OTHER SERVICES ARE
DEPLOYED ON DEMAND
AND AVAILABLE THROUGHT
TRAEFIK.

```
version: '3'

services:
  reverse-proxy:
    # The official v2.0 Traefik docker image
    image: traefik:v2.0
    # Enables the web UI and tells Traefik to listen to docker
    command: --api.insecure=true --providers.docker
    ports:
      # The HTTP port
      - "80:80"
      # The Web UI (enabled by --api.insecure=true)
      - "8080:8080"
    volumes:
      # So that Traefik can listen to the Docker events
      - /var/run/docker.sock:/var/run/docker.sock
```



```
# ...
whoami:
  # A container that exposes an API to show its IP address
  image: containous/whoami
  labels:
    - "traefik.http.routers.whoami.rule=Host(`whoami.docker.localhost`)"
```

```
docker-compose up -d --scale whoami=2
```

StripPrefix



Removing Prefixes From the Path Before Forwarding the Request

Remove the specified prefixes from the URL path.

Configuration Examples

[Docker](#)[Kubernetes](#)[Consul Catalog](#)[Marathon](#)[Rancher](#)[File \(TOML\)](#)[File \(YAML\)](#)

```
# Strip prefix /foobar and /fiibar
```

```
labels:
```

```
- "traefik.http.middlewares.test-stripprefix.stripprefix.prefixes=/foobar,/fiibar"
```



Configuration Options

General

The StripPrefix middleware will:

- strip the matching path prefix.
- store the matching path prefix in a `X-Forwarded-Prefix` header.