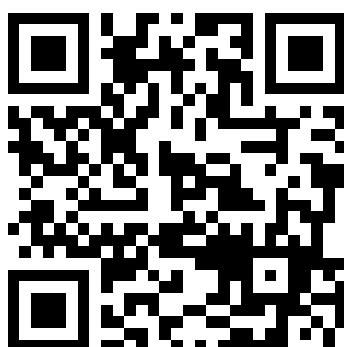


A La Découverte De Traefik 2.0 (⛵)



[sf≡ir]

Lille - 2019/06/13



<https://containous.github.io/slides/meetup-sfeir-lille-2019>

How To Use These Slides?

- **Browse the slides:** Use the arrows
 - Change chapter: Left/Right arrows
 - Next or previous slide: Top and bottom arrows
- **Overview of the slides:** keyboard's shortcut "o"
- **Speaker mode (and notes):** keyboard's shortcut "s"

Whoami

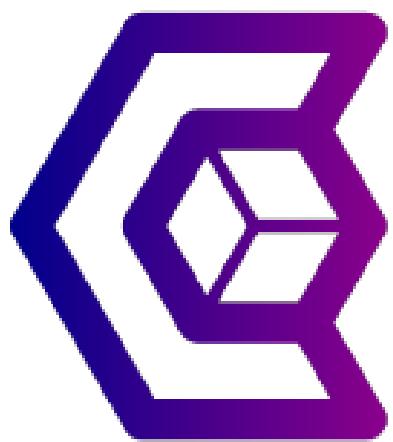
- Damien DUPORTAL:
 - Træfik's Developer  Advocate @ Containous
-  @DamienDuportal
-  dduortal



Containous

<https://containo.us>

- We Believe in Open Source
- We Deliver Traefik and Traefik Enterprise Edition
- Commercial Support
- 20 people, 90% tech

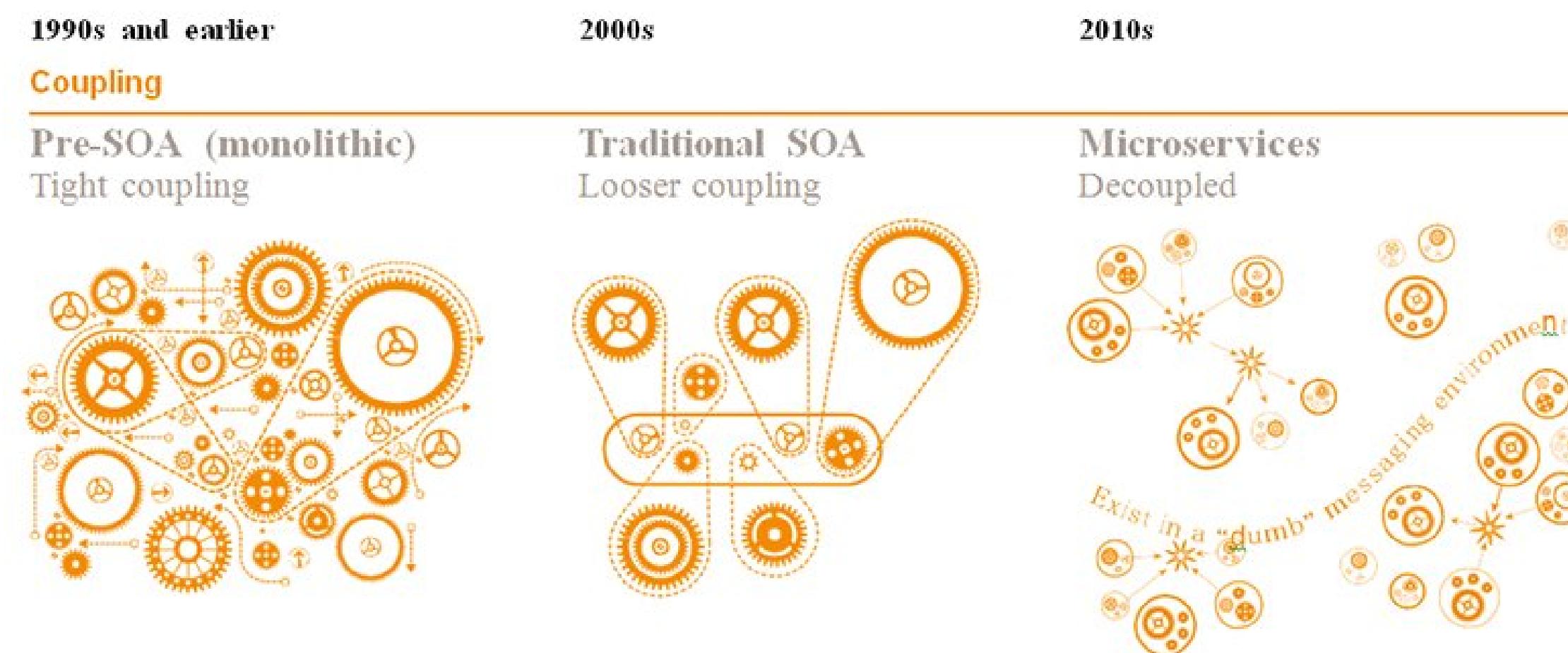


Why Traefik?



Why, Mr Anderson?

Evolution Of Software Design



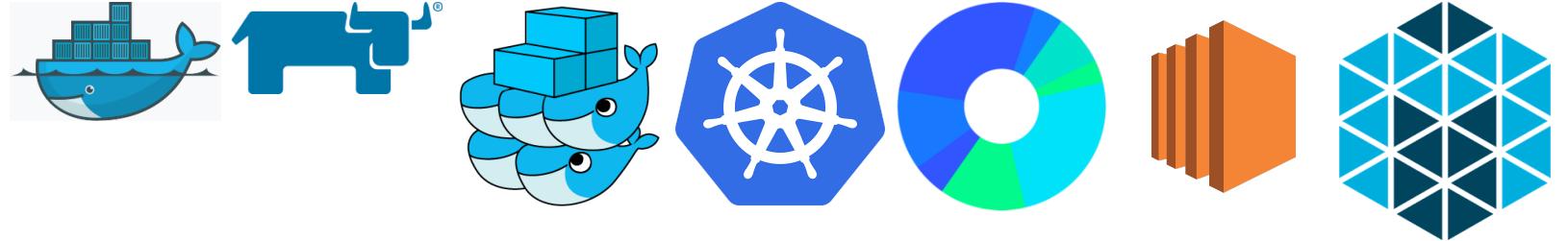
The Premise Of Microservices...



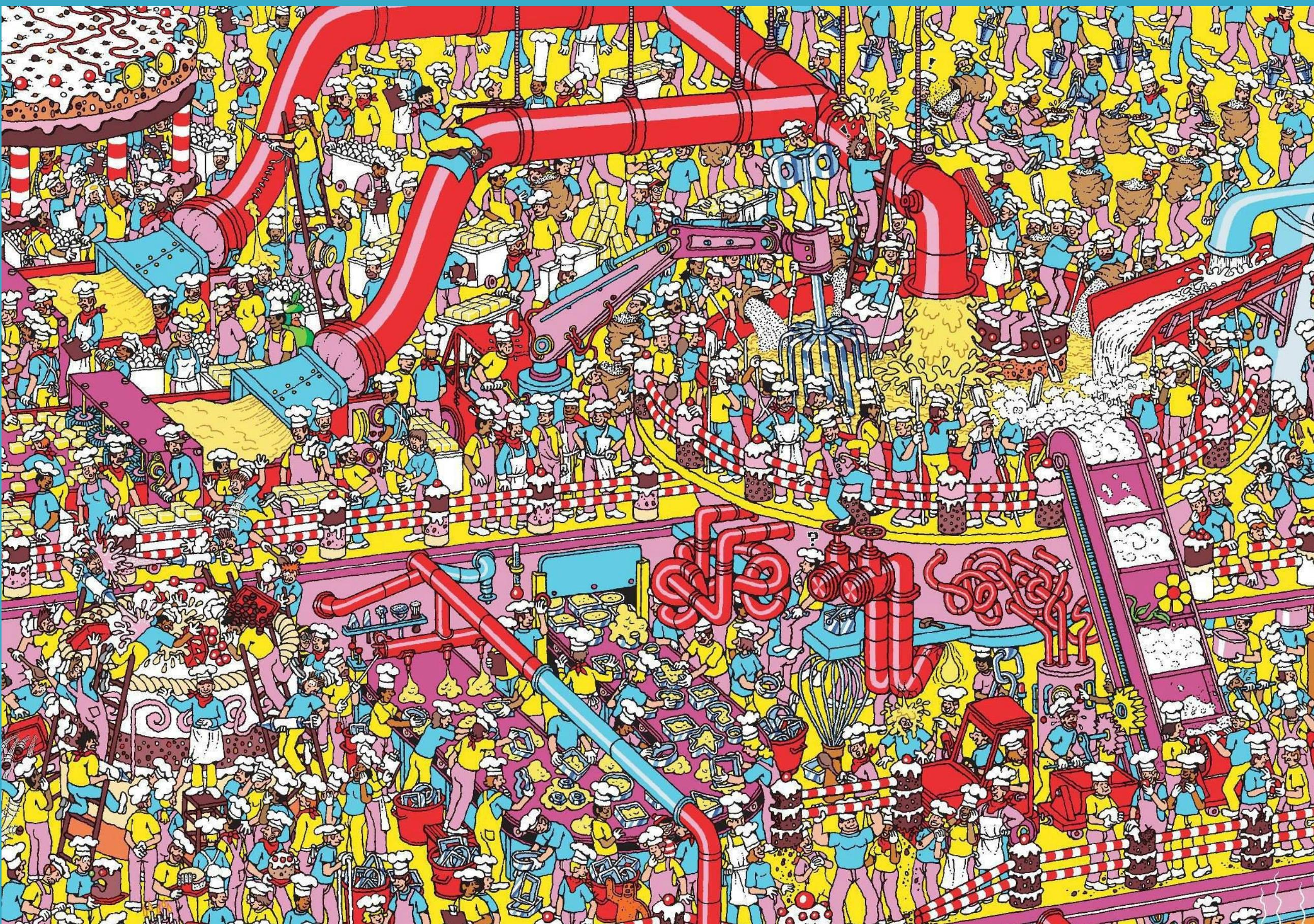
...And What Happens

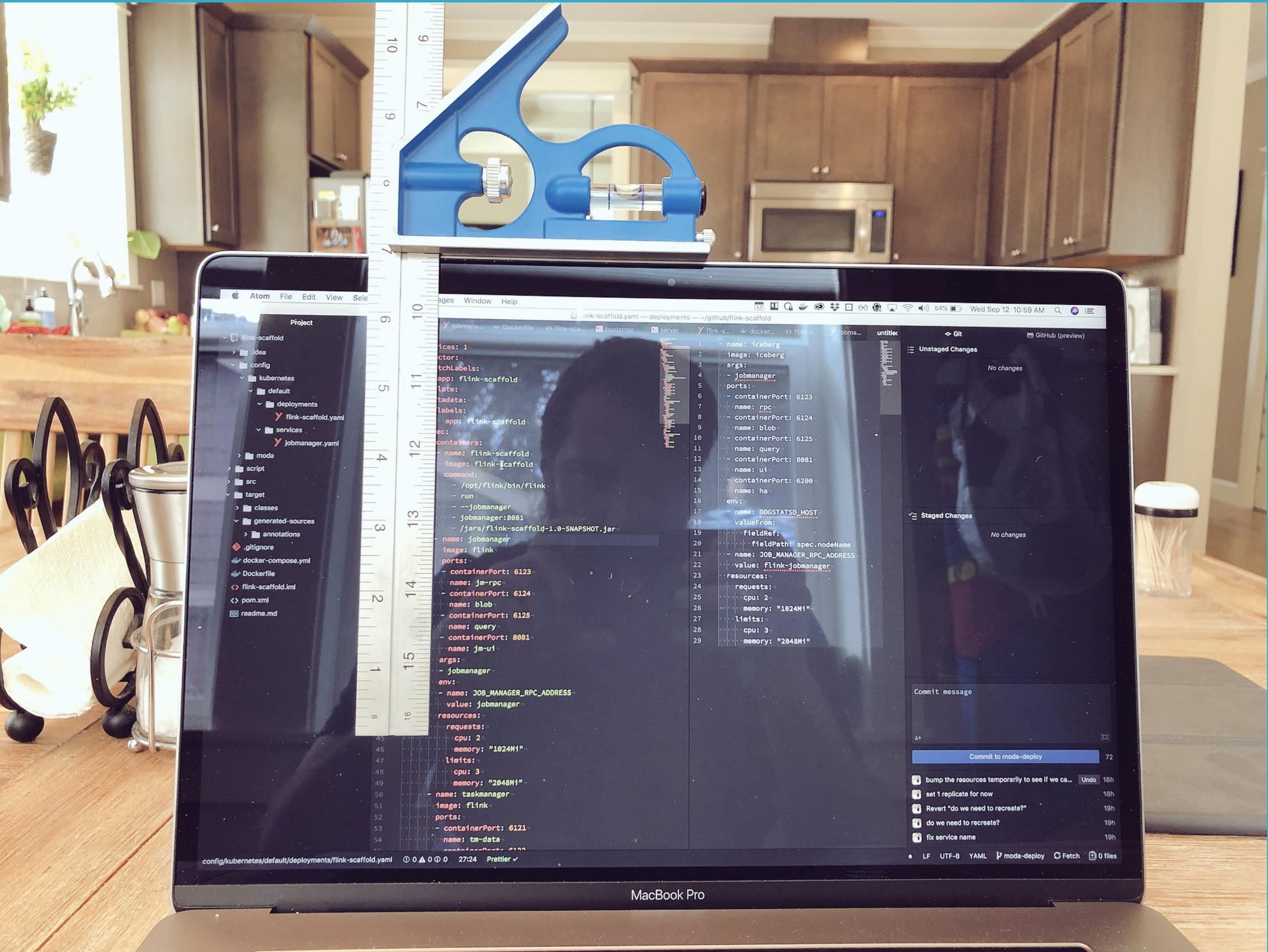


Tools Of The Trade



Where's My Service?





Source: <https://twitter.com/Caged/status/1039937162769096704>

What If I Told You?



That You Don't Have to Write This Configuration File...?

Here Comes Traefik!



Traefik Project

-  <https://github.com/containous/traefik>
- MIT License
- Written in Go
- 22,000+  750M+  350+ 
- Created in 2015
- Current stable branch: v1 . 7

BACK toTRAEFIK 2.0

Part →

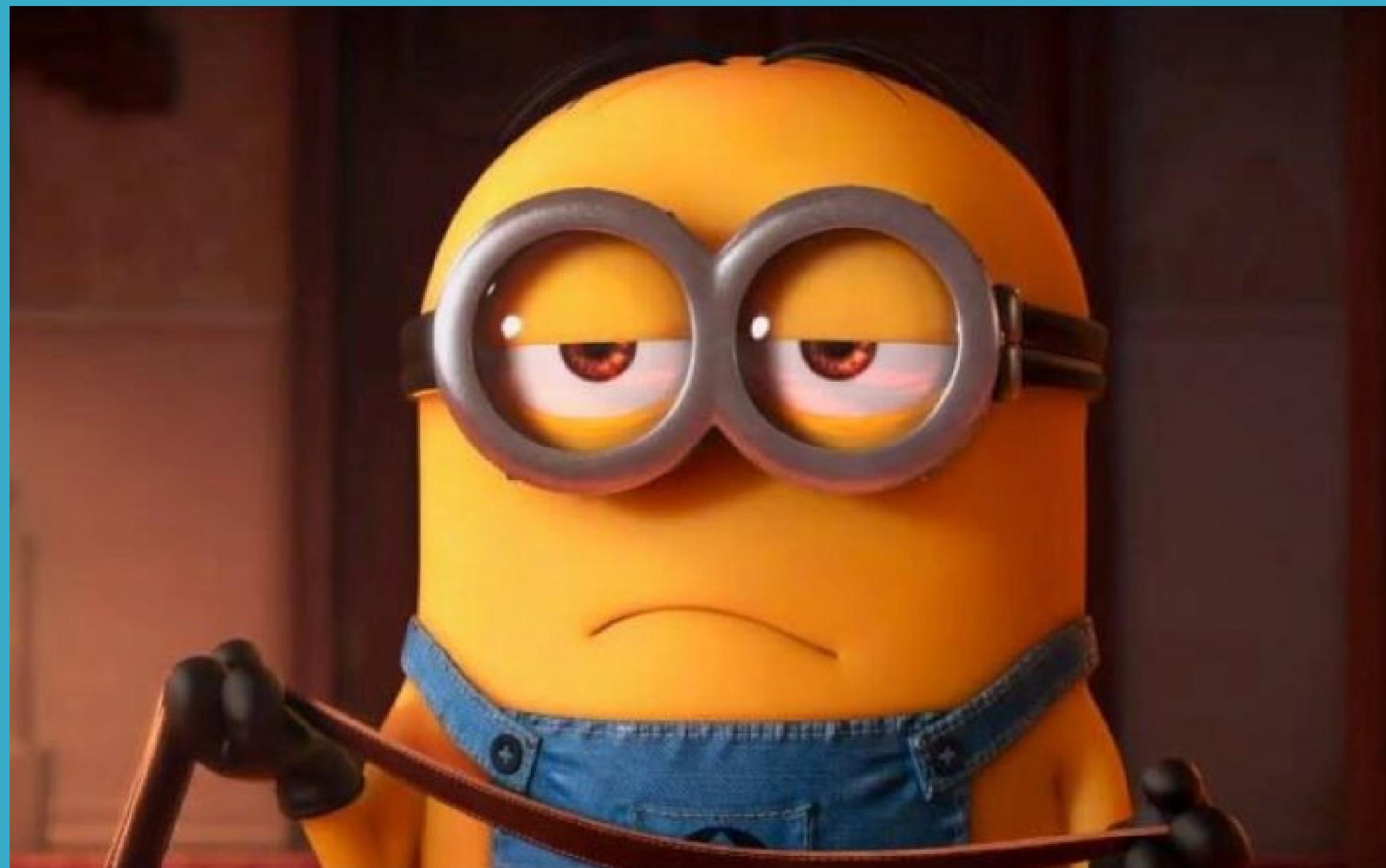


Traefik 2.0 Quick Overview

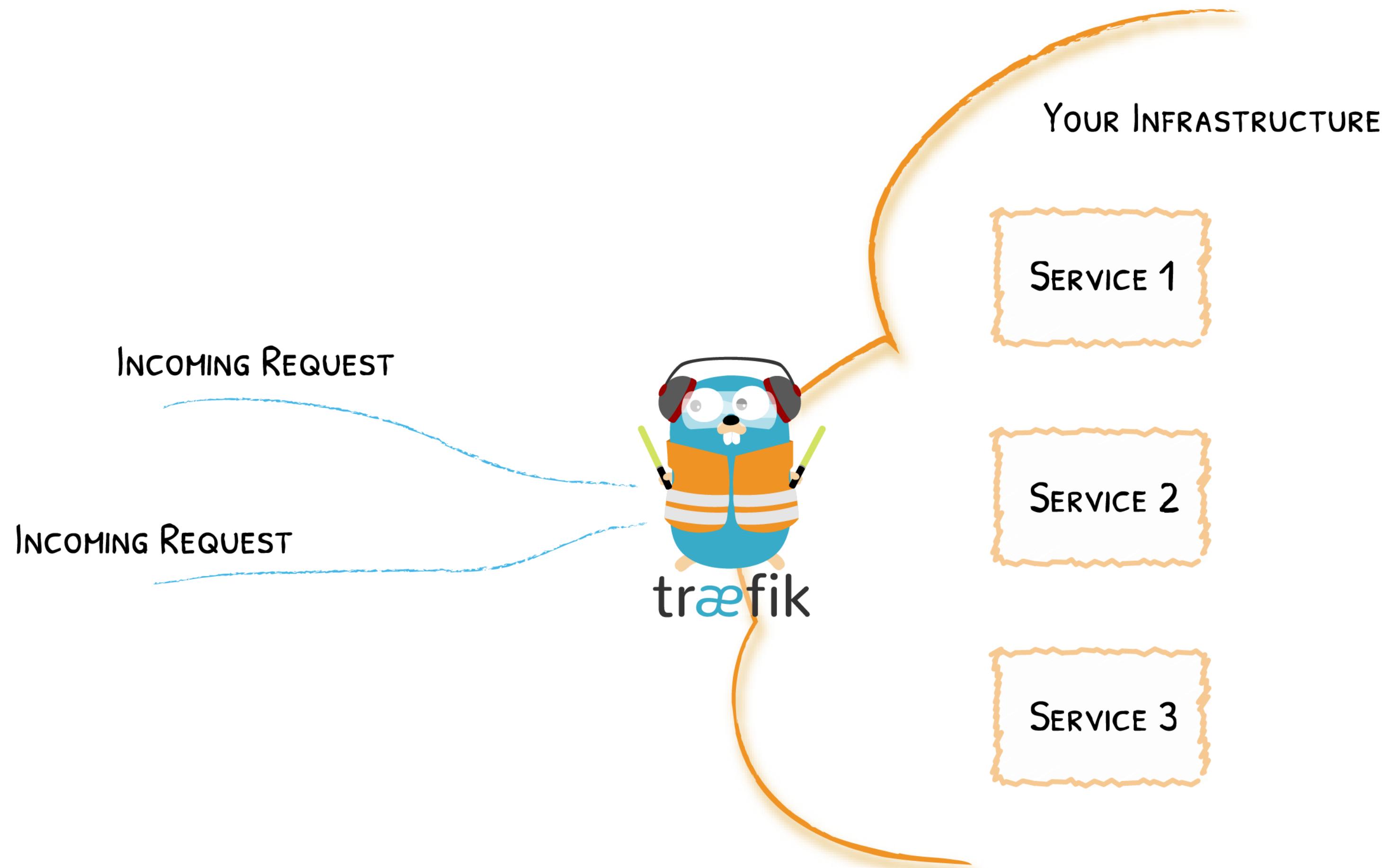
- Revamped Documentation
- Clarified Concepts
- Expressive Routing Rule Syntax
- Middlewares
- TCP Support
- And so Much More...

[Learn more on the blog post](#)

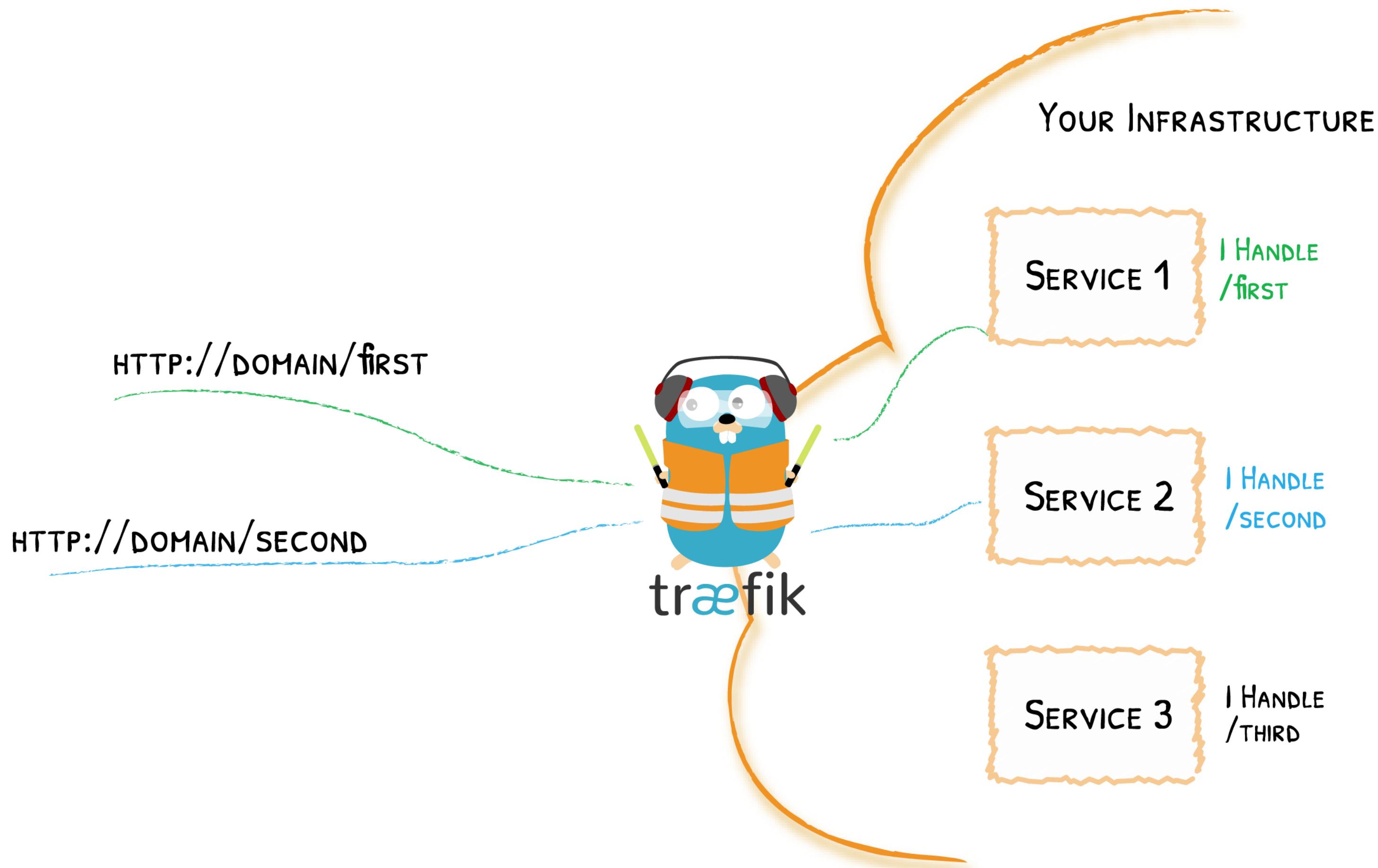
Traefik (V2.0) Core Concepts



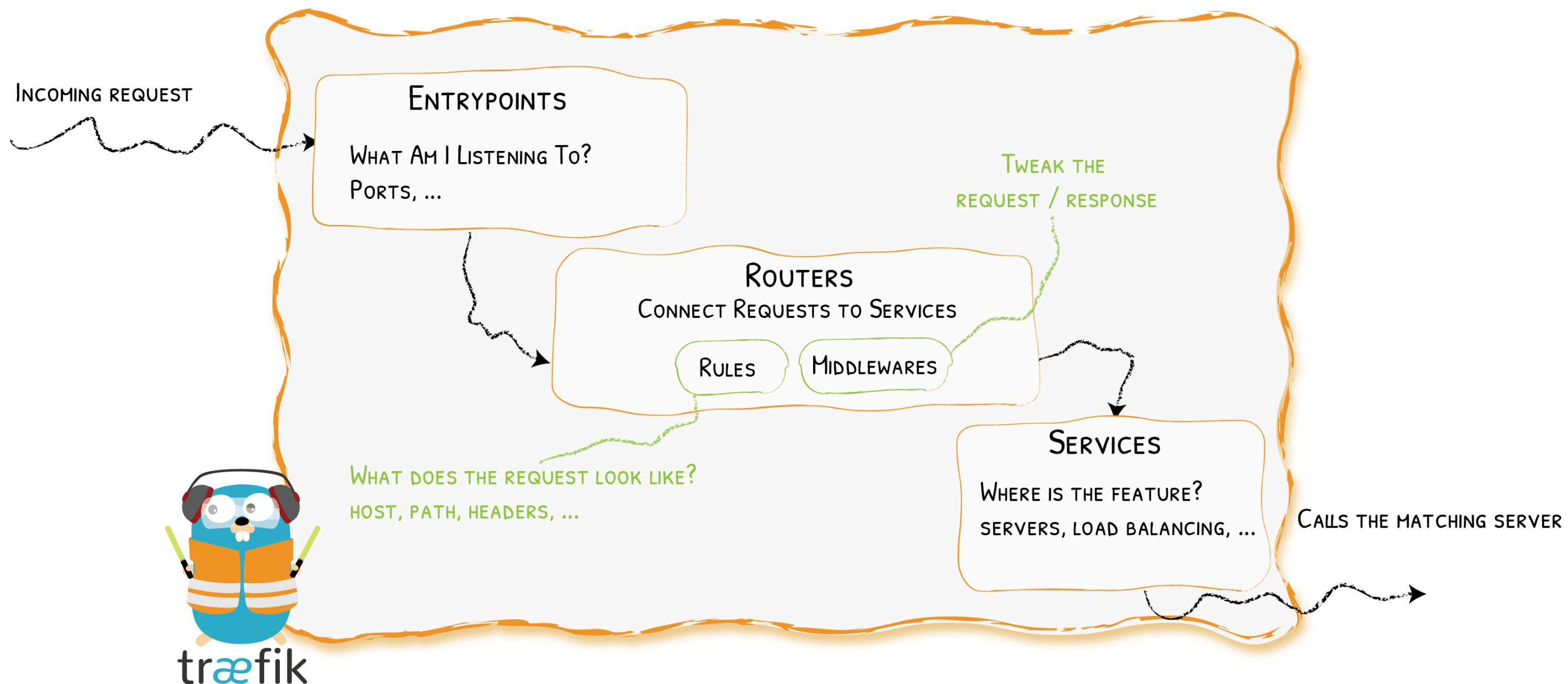
Traefik Is An Edge Router



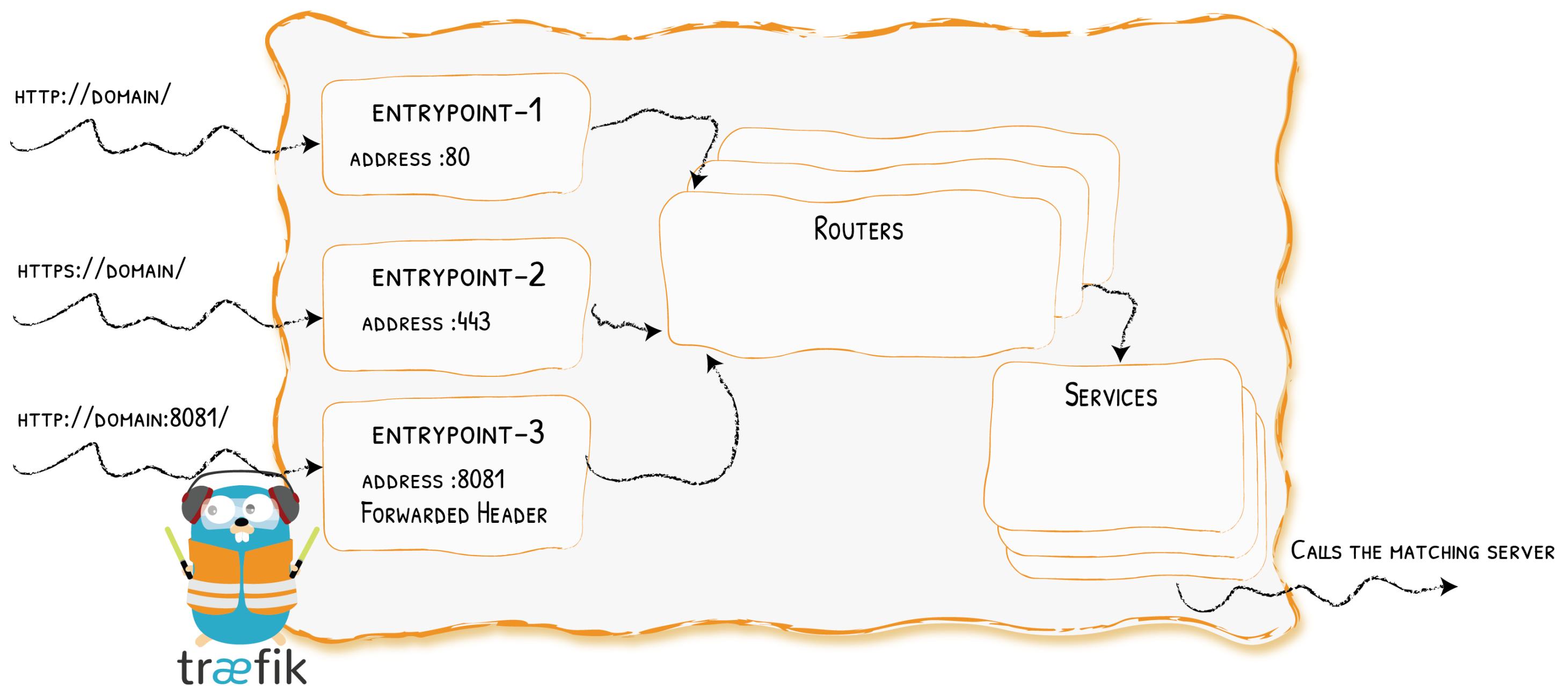
Traefik Dynamically Discovers Services



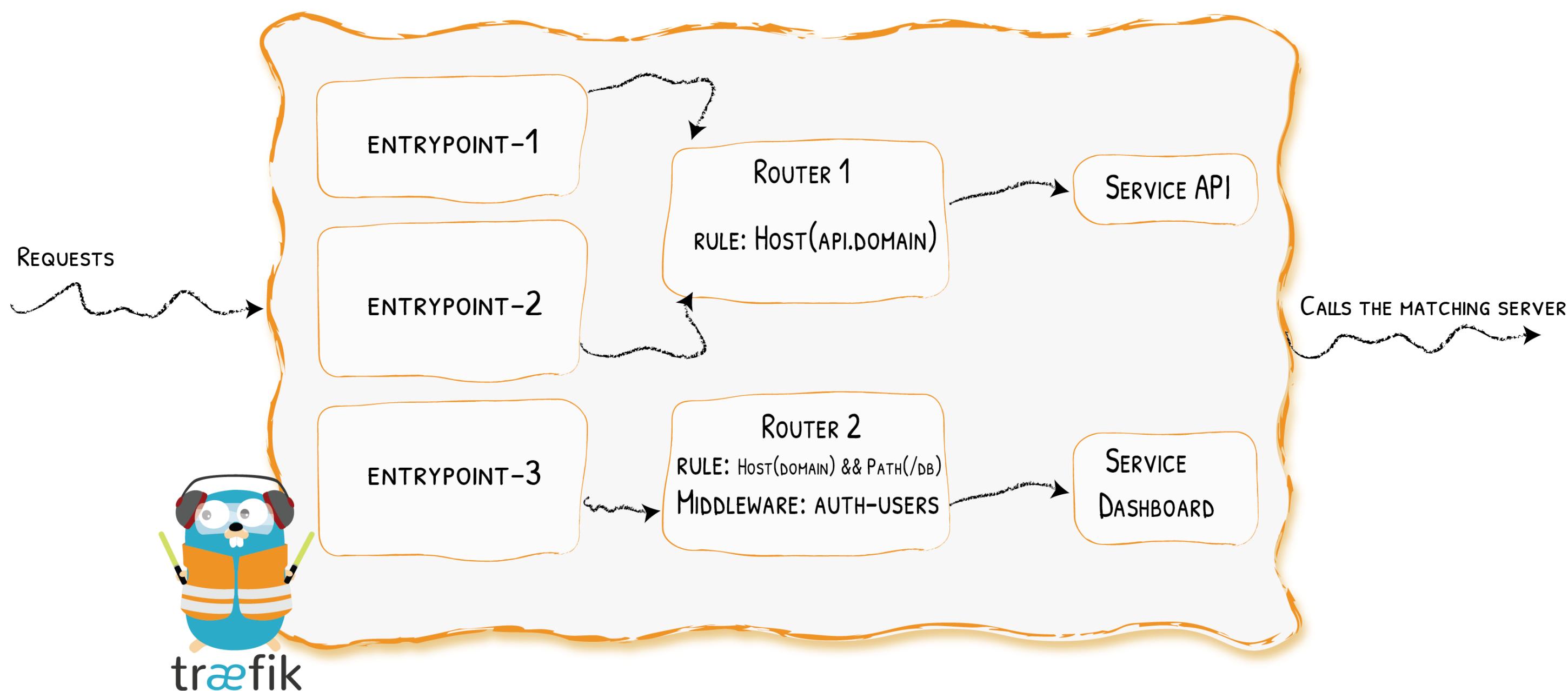
Architecture (V2.0) At A Glance



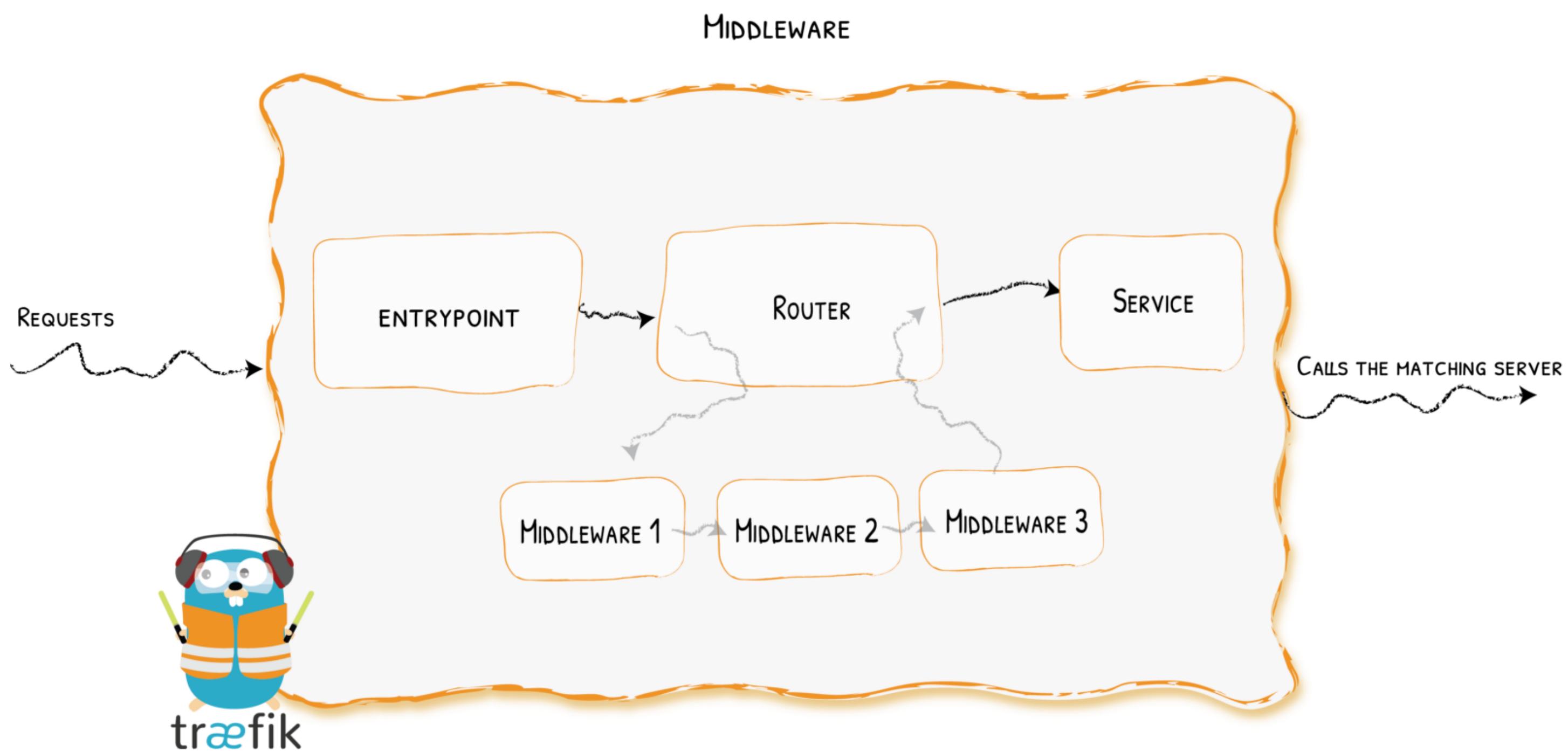
Entrypoints



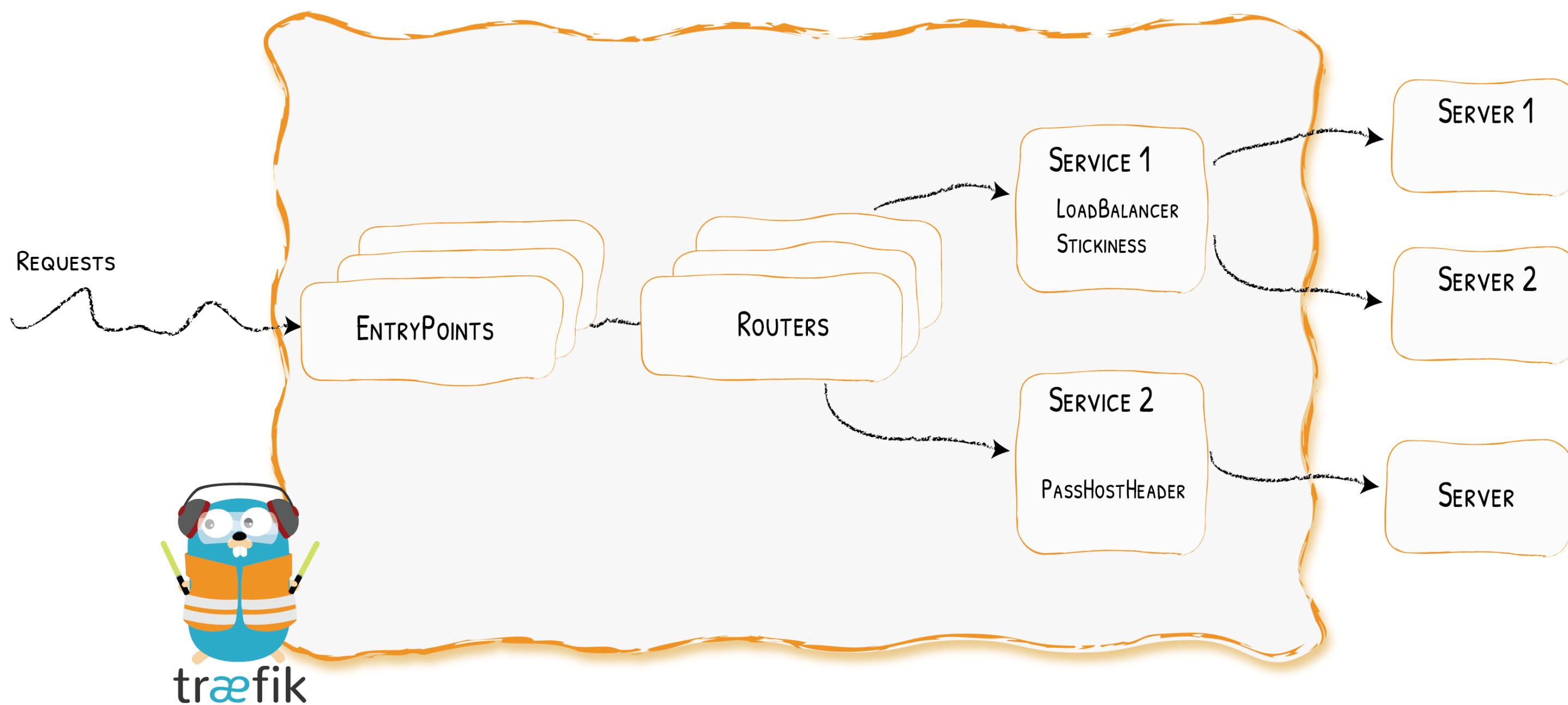
Routers



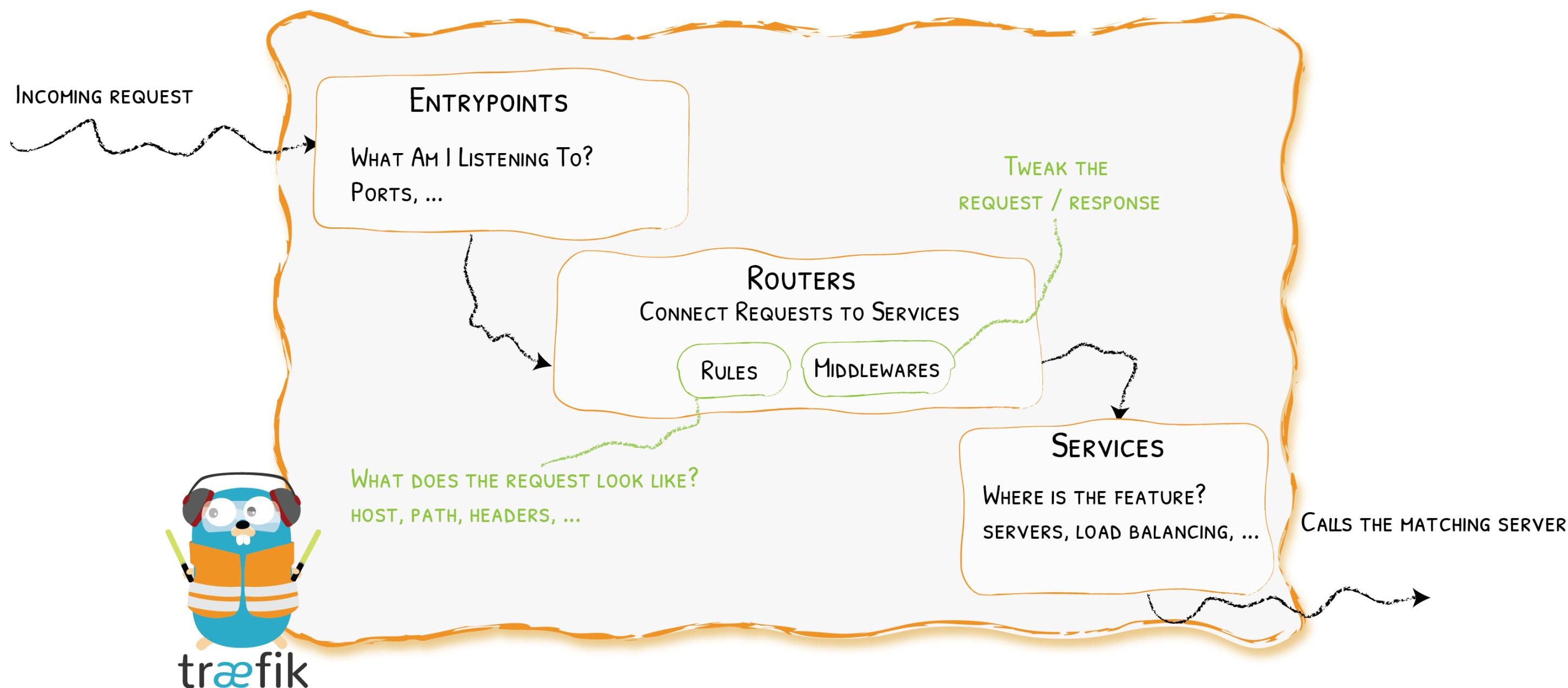
Middlewares



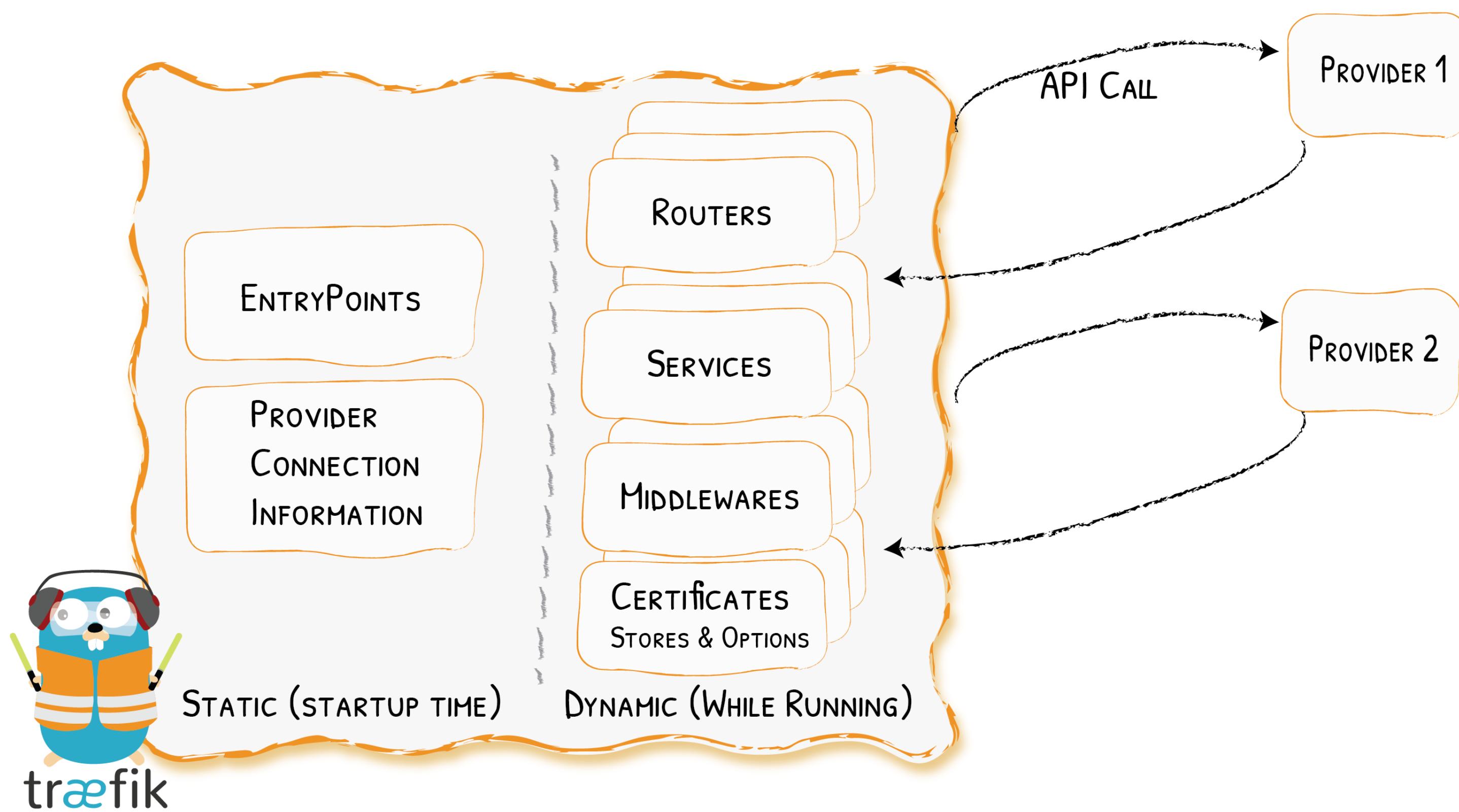
Services



Architecture (Again) At A Glance

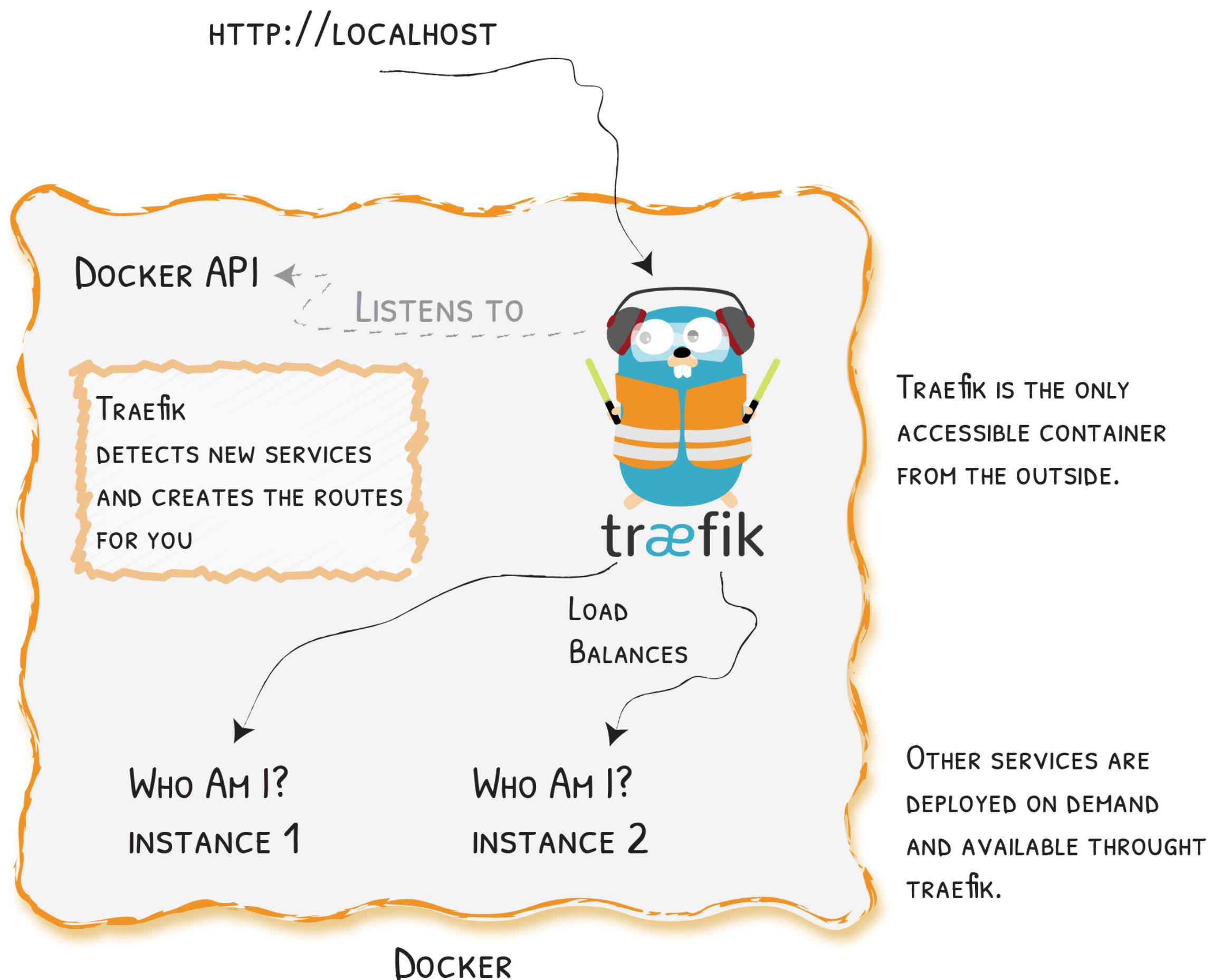


Static & Dynamic Configuration



Show Me The Configuration!

Simple Example With



With

- With Docker Compose:

```
version: '3'

services:
  reverse-proxy:
    image: traefik:v2.0
    command: --providers.docker
    ports:
      - "80:80"
    volumes:
      - /var/run/docker.sock:/var/run/docker.sock

  webapp:
    image: containous/whoami
    labels:
      - "traefik.http.routers.webapp.rule=Host(`localhost`)"
```

With Context

```
# https://mycompany.org/jenkins -> http://jenkins:8080/jenkins
jenkins:
  image: jenkins/jenkins:lts
  environment:
    - JENKINS_OPTS=--prefix=/jenkins
  labels:
    - "traefik.http.services.jenkins.LoadBalancer.server.Port=8080" # Because 50000 is also exposed
    - "traefik.http.routers.jenkins.rule=Host(`mycompany.org`) && PathPrefix(`/jenkins`)"
    - "traefik.http.routers.jenkins.service=jenkins"
```

With 🐠: Rewrites

```
# https://mycompany.org/gitserver -> http://gitserver:3000/
gitserver:
  image: gitea/gitea
  labels:
    - "traefik.http.routers.gitserver.rule=Host(`mycompany.org`) && PathPrefix(`/gitserver`)"
    - "traefik.http.middlewares.gitserver-striprefix.striprefix.prefixes=/gitserver"
    - "traefik.http.routers.gitserver.middlewares=gitserver-striprefix"
```

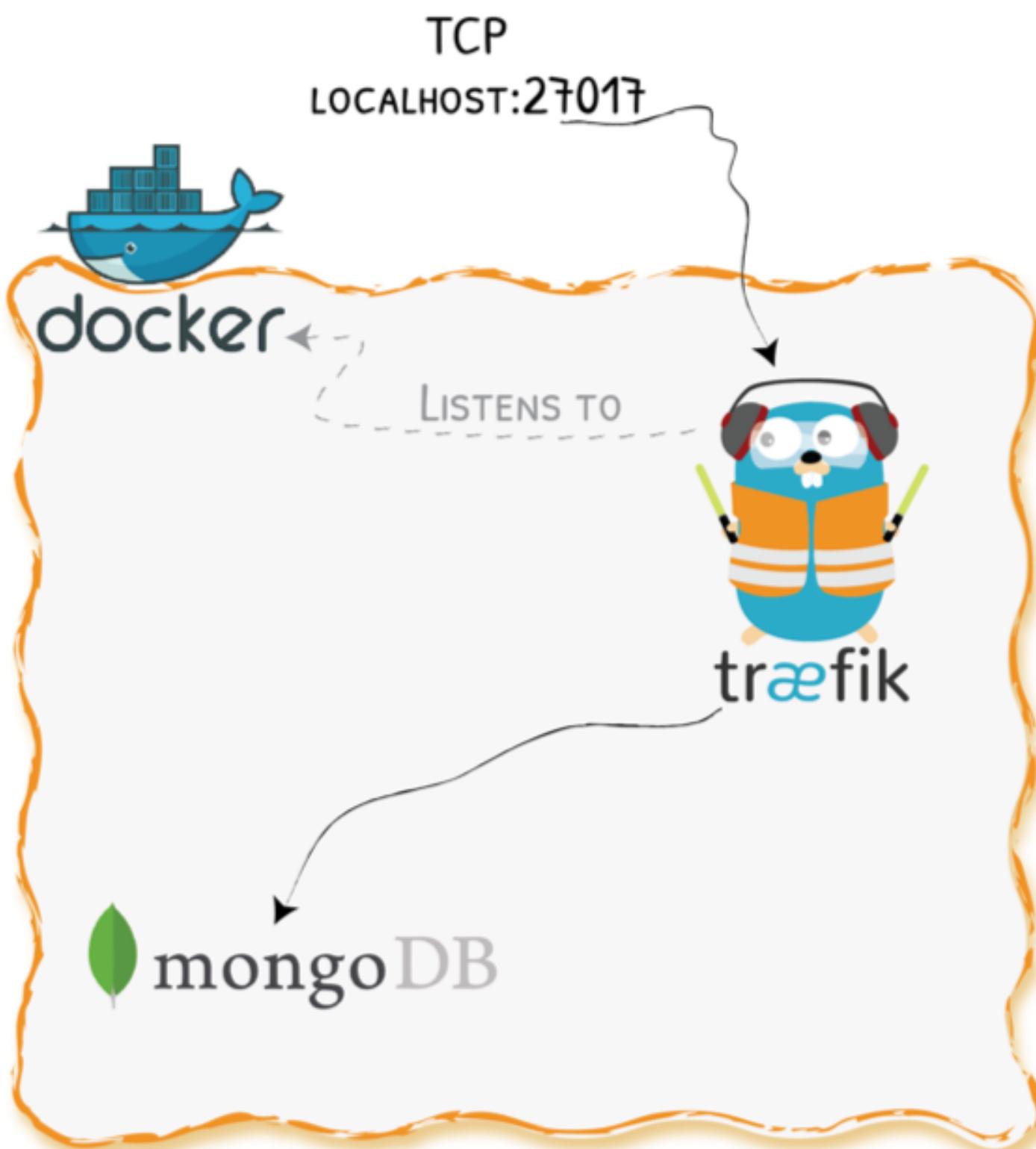
With Websockets

```
# https://mycompany.org/webterminal -> http://webterminal/
webterminal:
  image: ts10922/ttyd
  labels:
    - "traefik.http.routers.devbox.rule=Host(`mycompany.org`) && PathPrefix(`/webterminal`)"
```



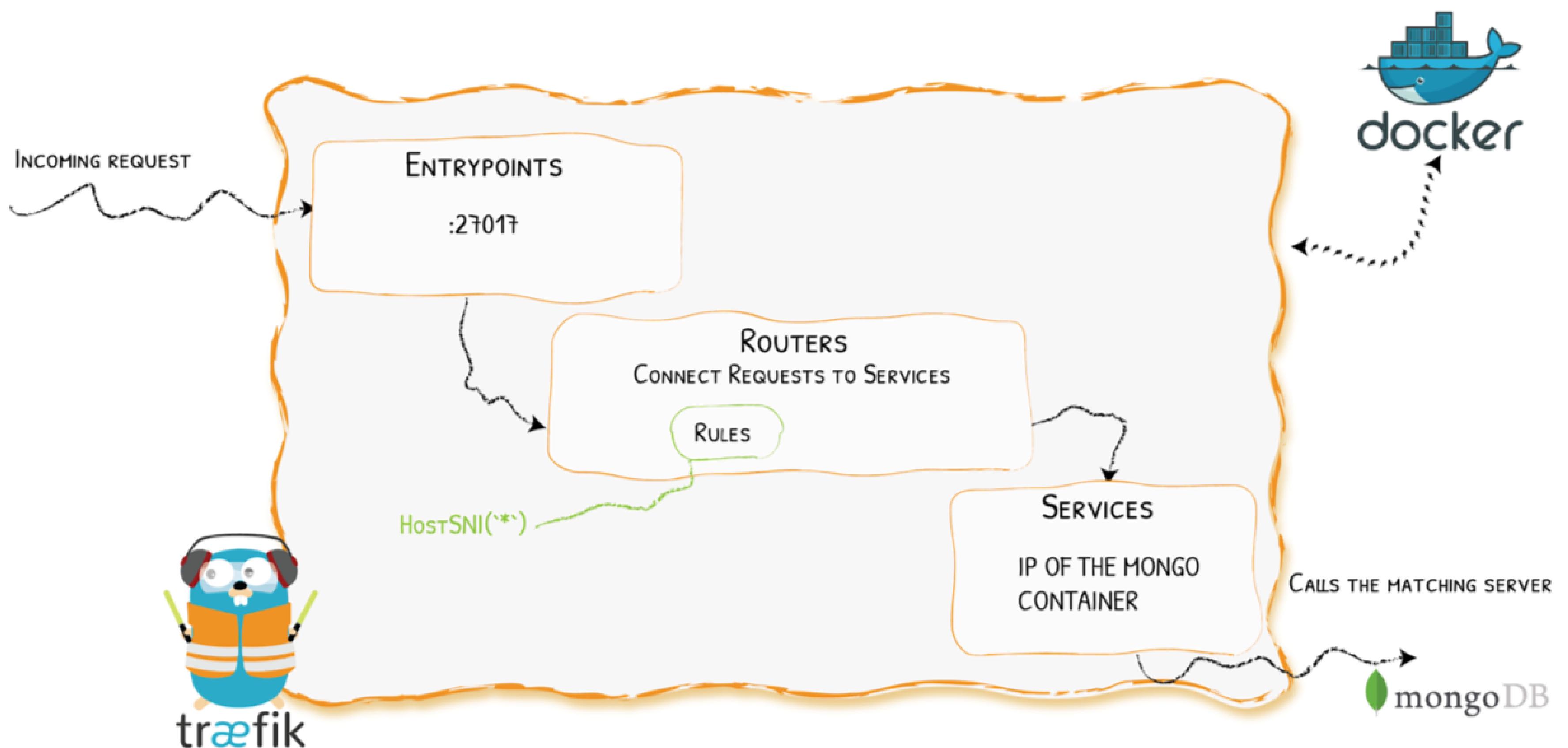
HTTP
&
TCP

Demo 1 - Straightforward TCP Routing

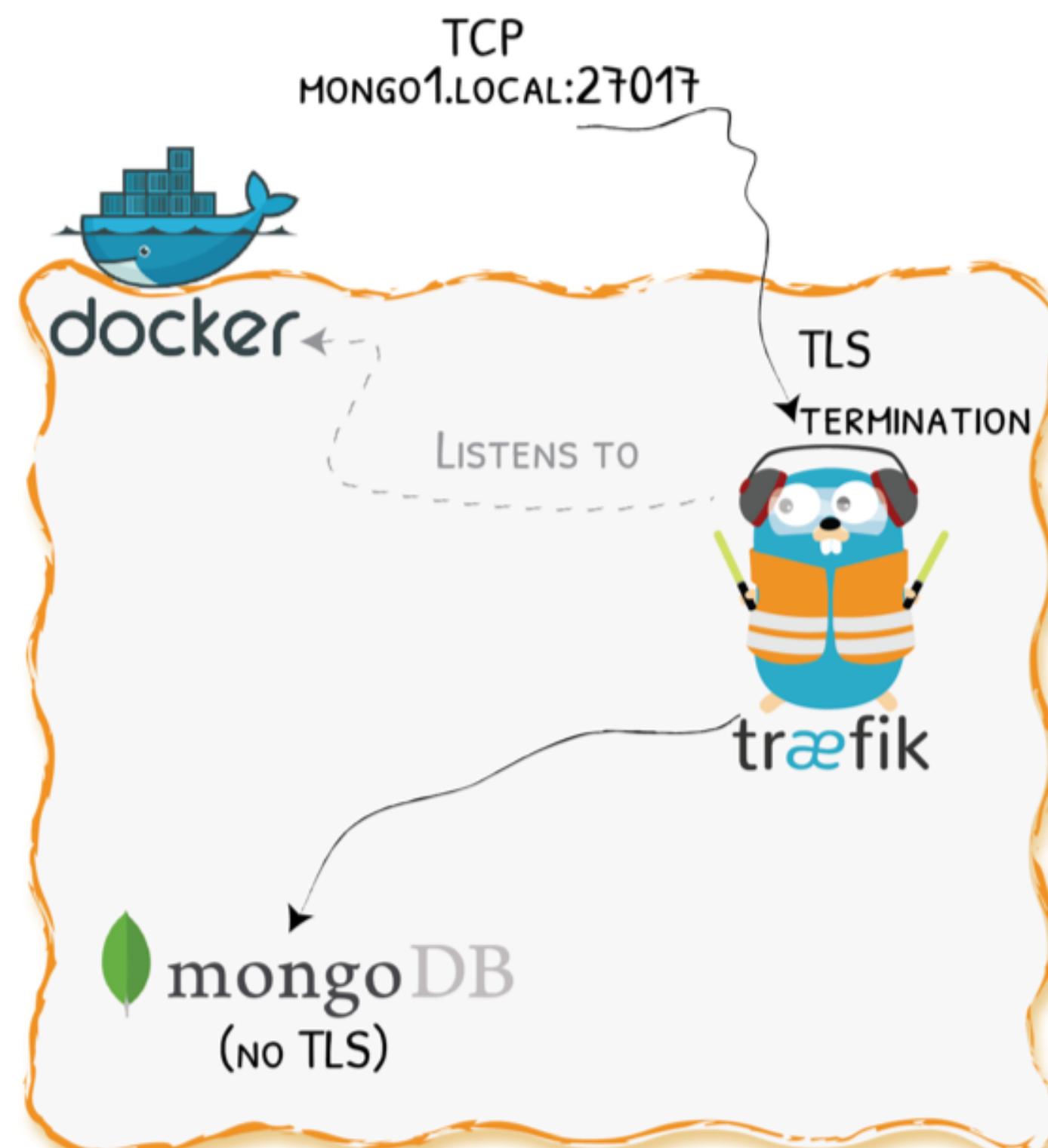


Demo Code on [GitHub](#)

Demo 1 - Configuration

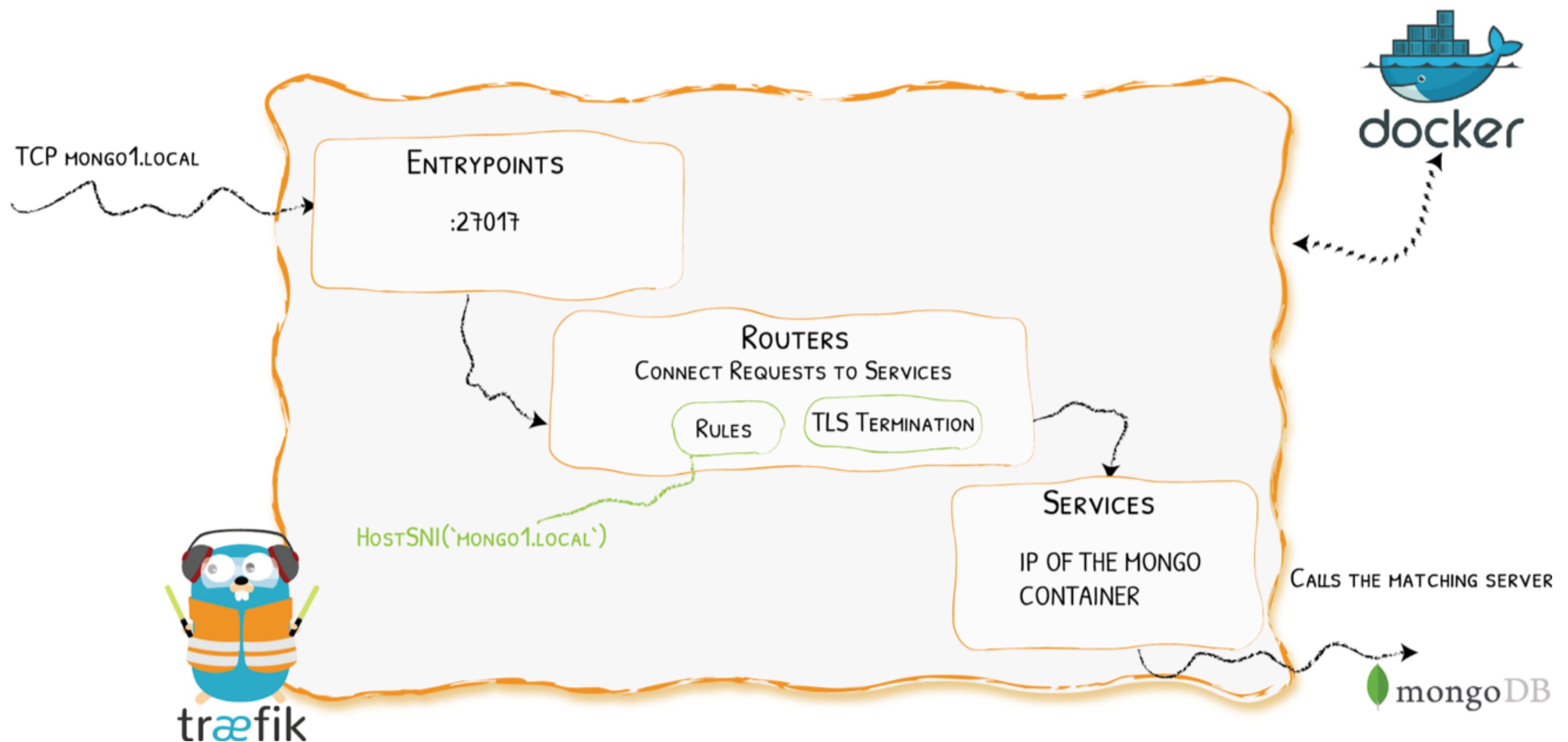


Demo 2 - Let's Add TLS To TCP With Traefik

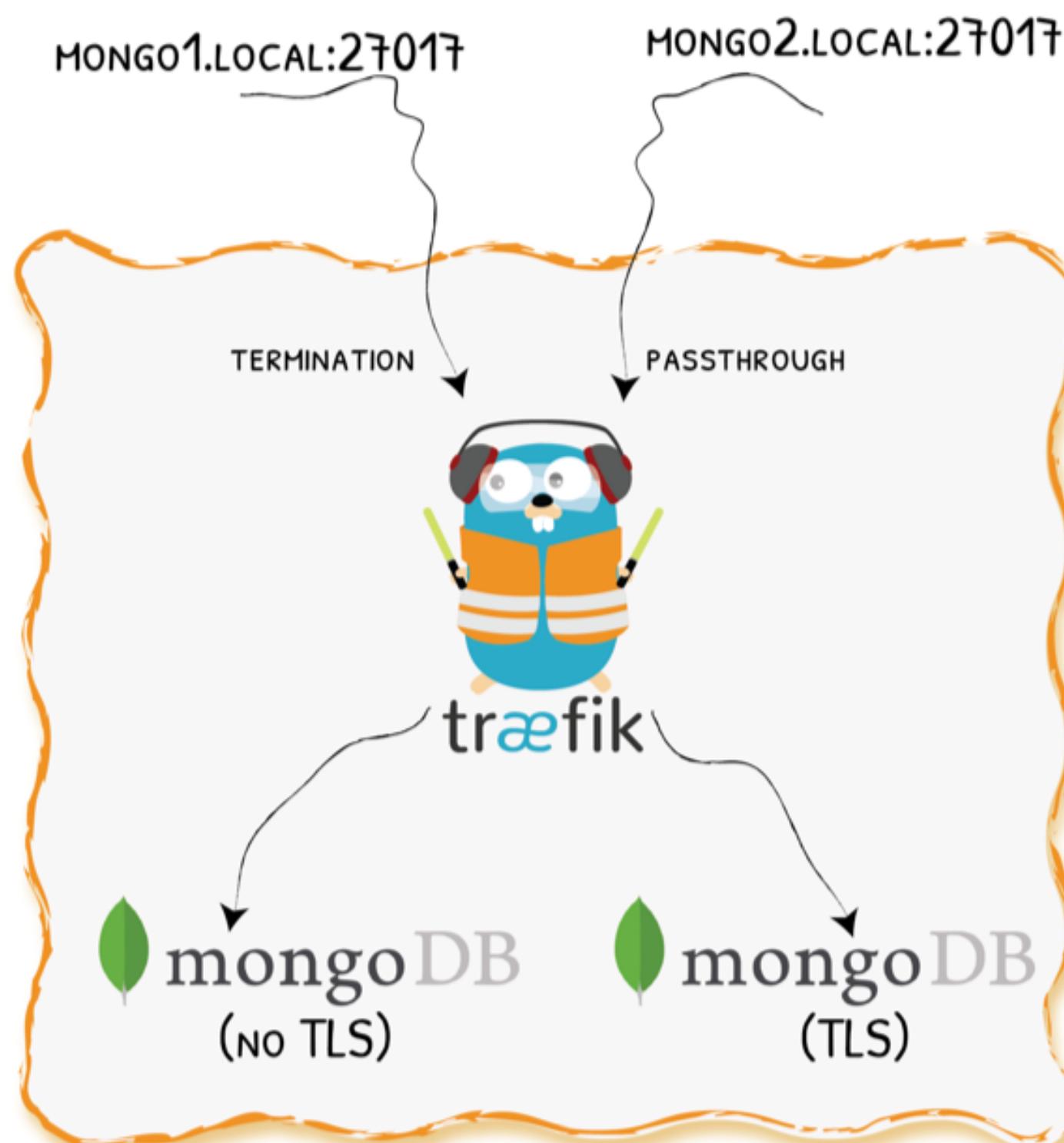


Demo Code on [GitHub](#)

Demo 2 - Configuration

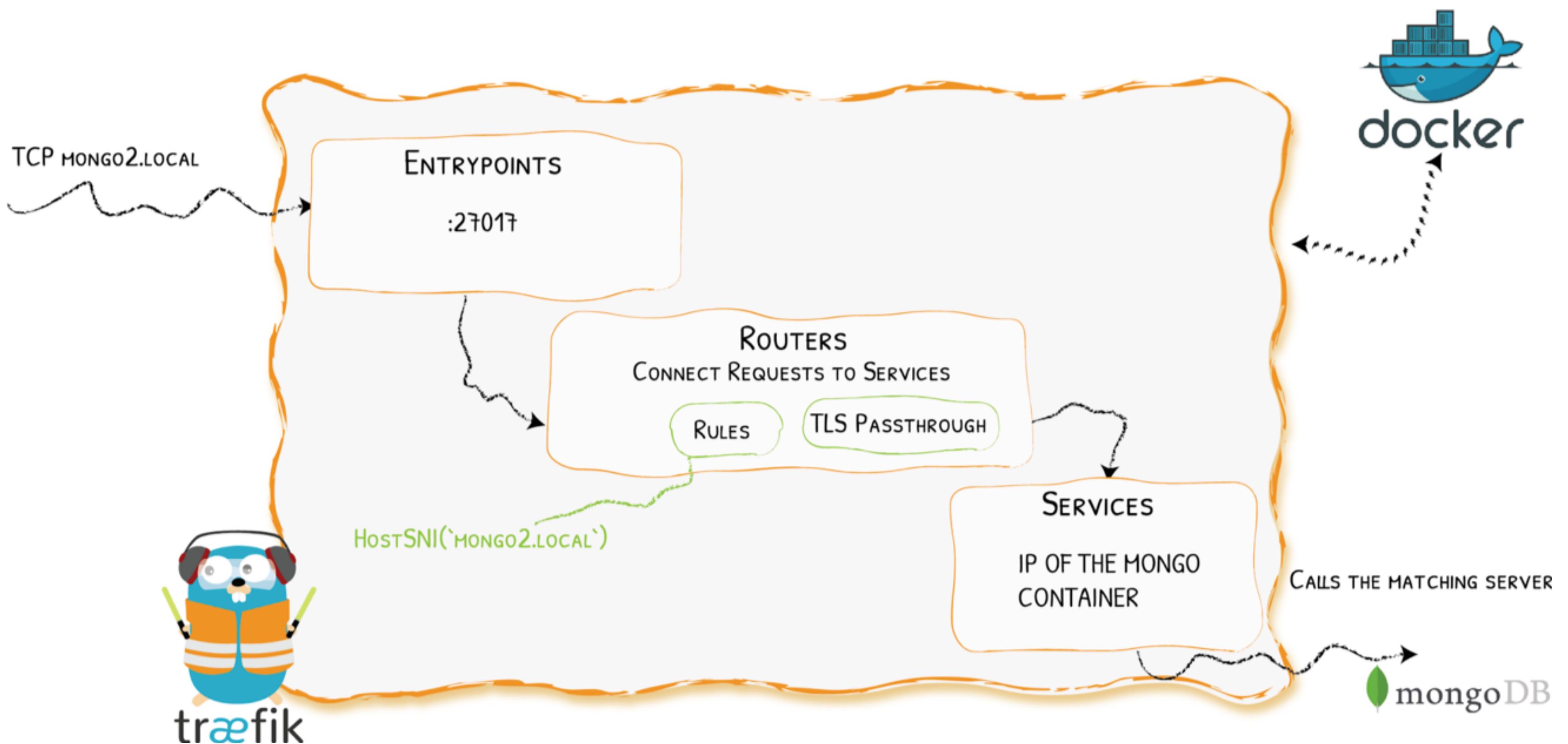


Demo 3 - SNI Routing + TLS Passthrough

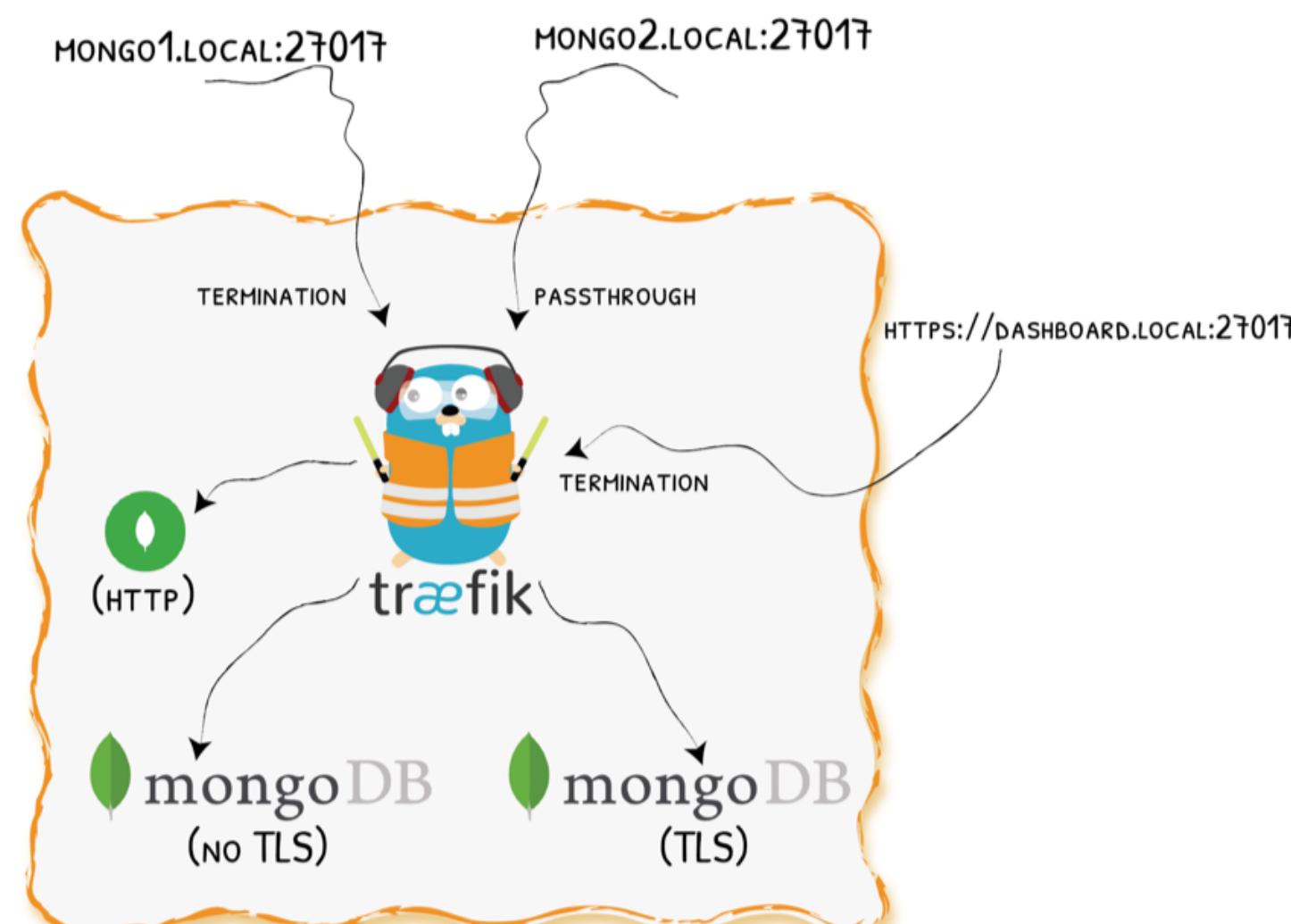


Demo Code on [GitHub](#)

Demo 3 - Configuration



Demo 4 - Muxing HTTPS And TCP On The Same Port



Demo Code on [GitHub](#)

Traefik With ⚓

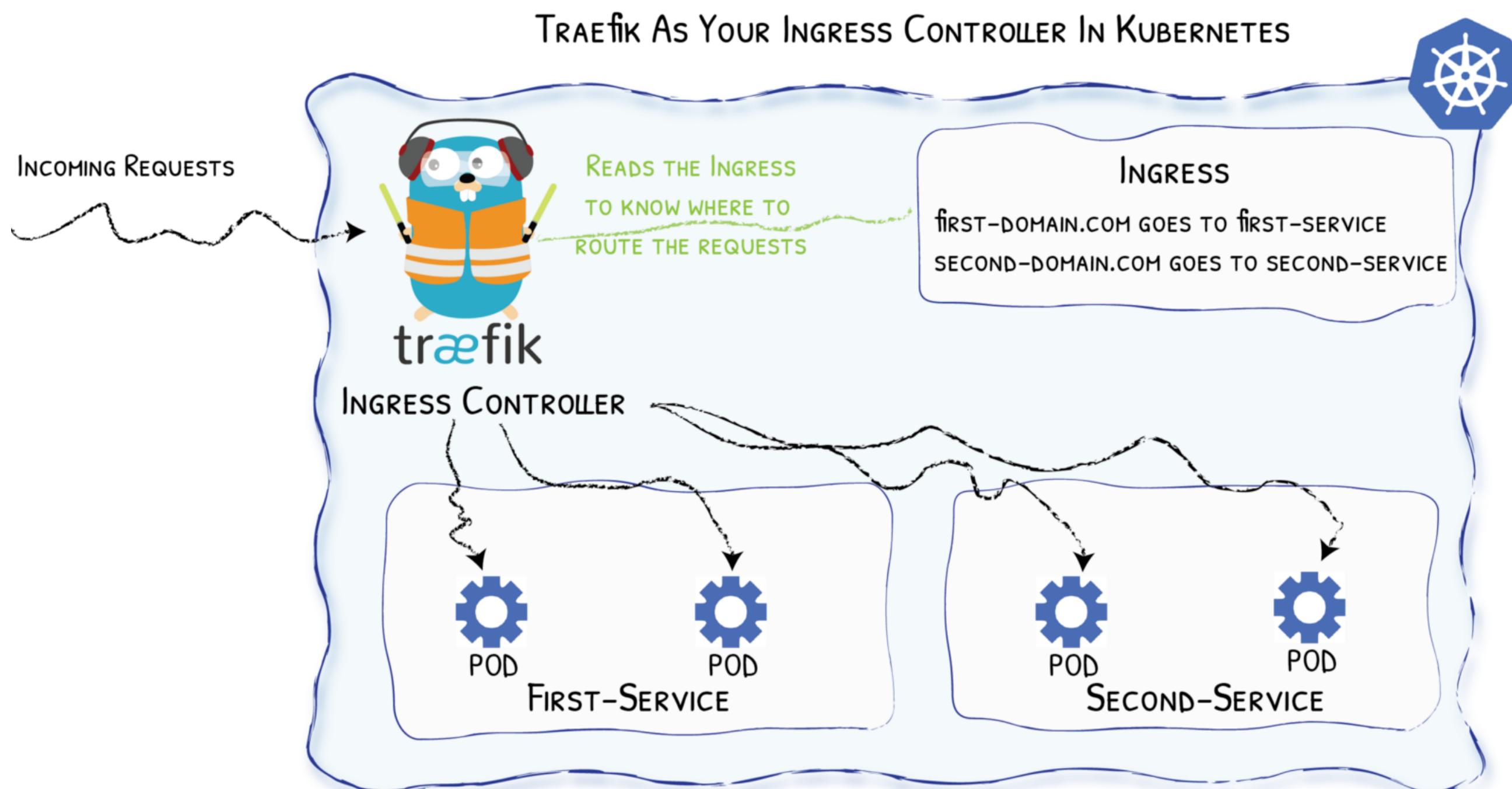


Diagram from <https://medium.com/@geraldcroes>

Example Code With ⚓

```
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  annotations:
    kubernetes.io/ingress.class: 'traefik'
spec:
  rules:
  - host: localhost
    http:
      paths:
      - path: "/whoami"
        backend:
          serviceName: webapp
          servicePort: 80
```

✳️ CRD - Custom Resources Definition

```
apiVersion: traefik.containo.us/v1alpha1
kind: IngressRoute
spec:
  entrypoints:
    - web
    - web-secure
  routes:
    - match: Host(`traefik.io`) && PathPrefix(`/foo`)
      kind: Rule
      services:
        - name: whoami1
          port: 80
          strategy: RoundRobin
      middlewares:
        - name: stripprefix
    - match: Host(`containo.us`) && Method(`POST`)
      kind: Rule
      services:
        - name: whoami2
          port: 80
  tls:
    secretName: supersecret
```

More To Come For V2.0

- New WebUI
- Newmetrics
- UDP
- YAML
- TLS stores & options
- Canary

More Info

bit.ly/traefik-v2

We Also Missed Talking About...

A cloud of technology terms in various colors, including:

- MESOS
- ZIPKIN
- LIMITING
- KUBERNETES
- Dynamic Metrics
- HTTP ERROR
- CERTIFICATE
- TLS Reverse-Proxy
- HEADERS
- GRPC
- S3 BUCKET
- CIRCUIT BREAKERS
- DYNAMIC/WILDCARD
- Security Configurations
- Tracing PROXY
- SECRETS
- PROMETHEUS
- JAEGER
- WEBSOCKETS
- SSL
- FORWARDING
- REDIRECTS
- DOCKER
- PROTOCOL
- CHECKS
- CLUSTER AUTH
- HSTS
- RATE
- CONSUL
- SWARM MODE
- HEALTH

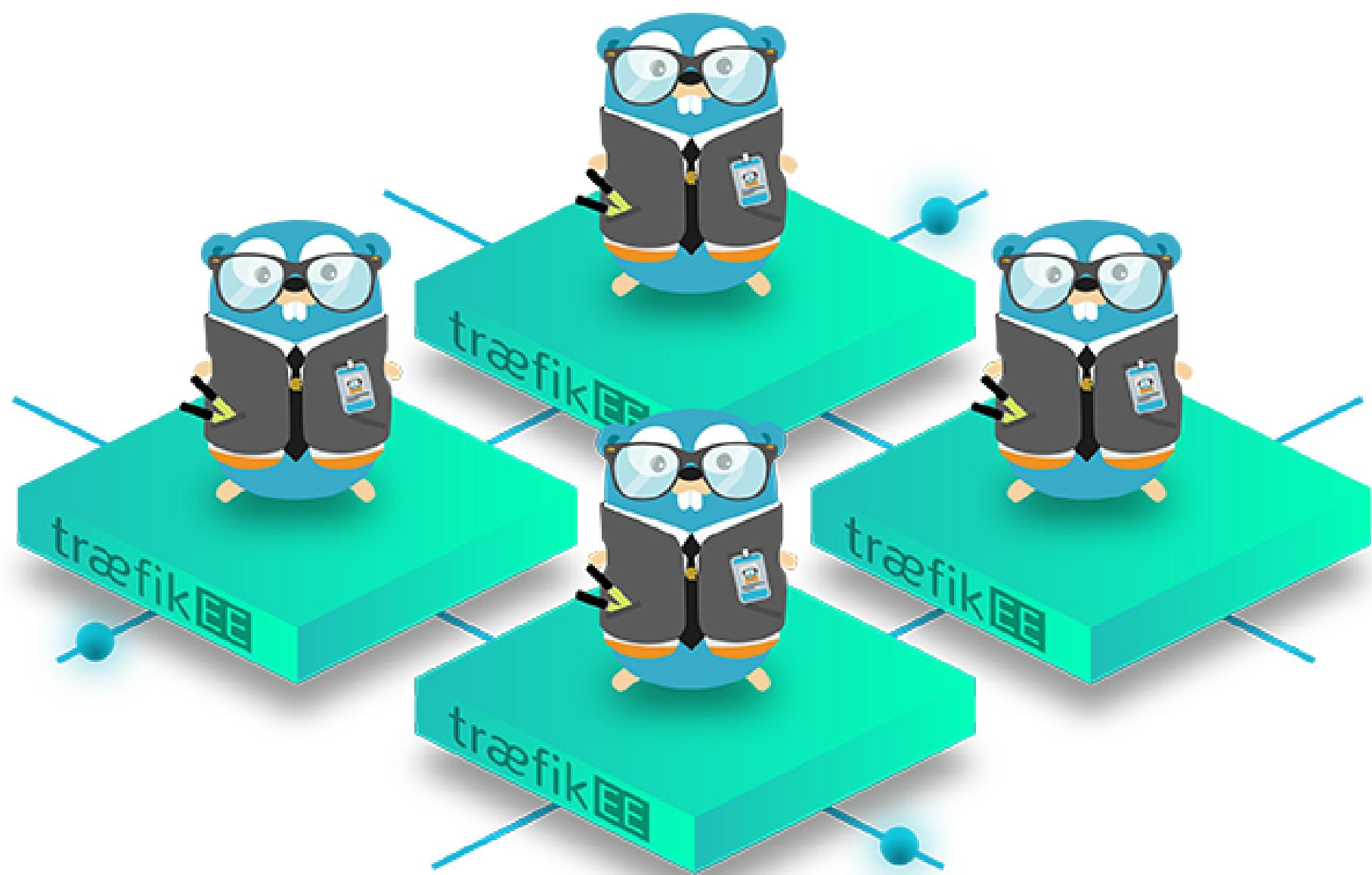
**TO BE
CONTINUED...**

The Herd



You came to the wrong neighbour

Traefik Comes In Herd



High Availability



HIGH AVAILABILITY

traefik ENTERPRISE EDITION



SECURITY

traefik ENTERPRISE EDITION

Scalability

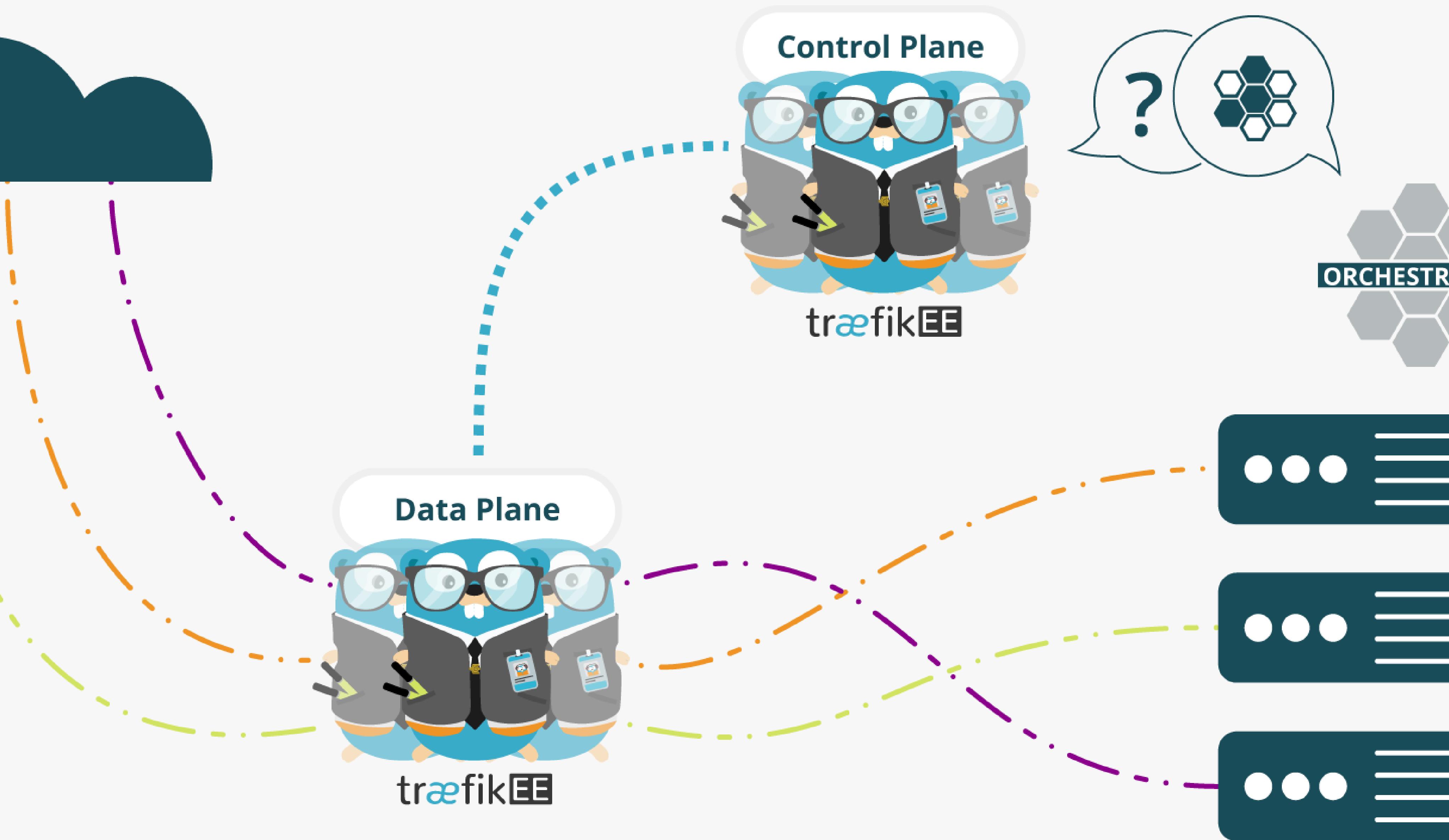


SCALABILITY

traefik ENTERPRISE EDITION

INTERNET

TO YOUR INFRA



As Simple As Traefik

- Install it:

```
# Cluster Installation
traefikeectl install \
  --licensekey="SuperSecretLicence" \
  --dashboard \
  --kubernetes # Or --swarm
```

- Configure it:

```
# Routing Configuration, same as Traefik's
traefikeectl deploy \
  --acme.email=ssl-admin@mycompany.org
  --acme.tlsChallenge
  ...
```

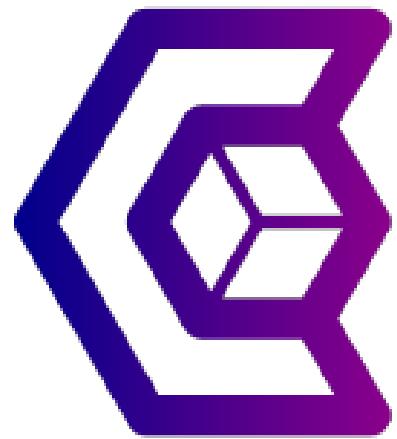
Early (Free) Access

<https://containo.us/traefikee>

But
What About Open Source?



We Are Hiring!

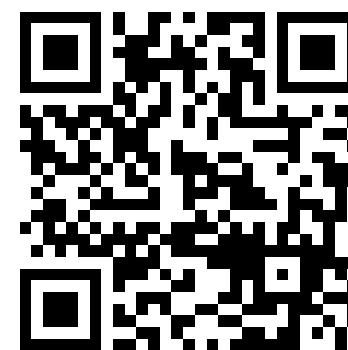


```
docker run -it containous/jobs
```

Thank You!

 @DamienDuportal

 dduportal



- Slides (HTML): <https://containous.github.io/slides/meetup-sfeir-lille-2019>
- Slides (PDF): <https://containous.github.io/slides/meetup-sfeir-lille-2019/slides.pdf>
- Source on : <https://github.com/containous/slides/tree/meetup-sfeir-lille-2019>