Smart Routing Et HTTPS Pour Tous

Traefik En Action !

https://containous.github.io/slides/bbl-decathlon-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"

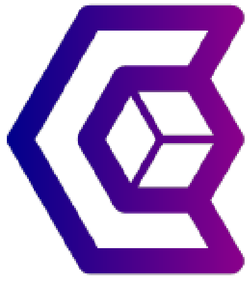
Whoami

- Damien DUPORTAL:
  - Træfik's Developer 🥑 Advocate @ Containous
- 🐦 @DamienDuportal
- dduportal

Containous

https://containo.us

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



Why Traefik?

Why, Mr Anderson?

Evolution Of Software Design

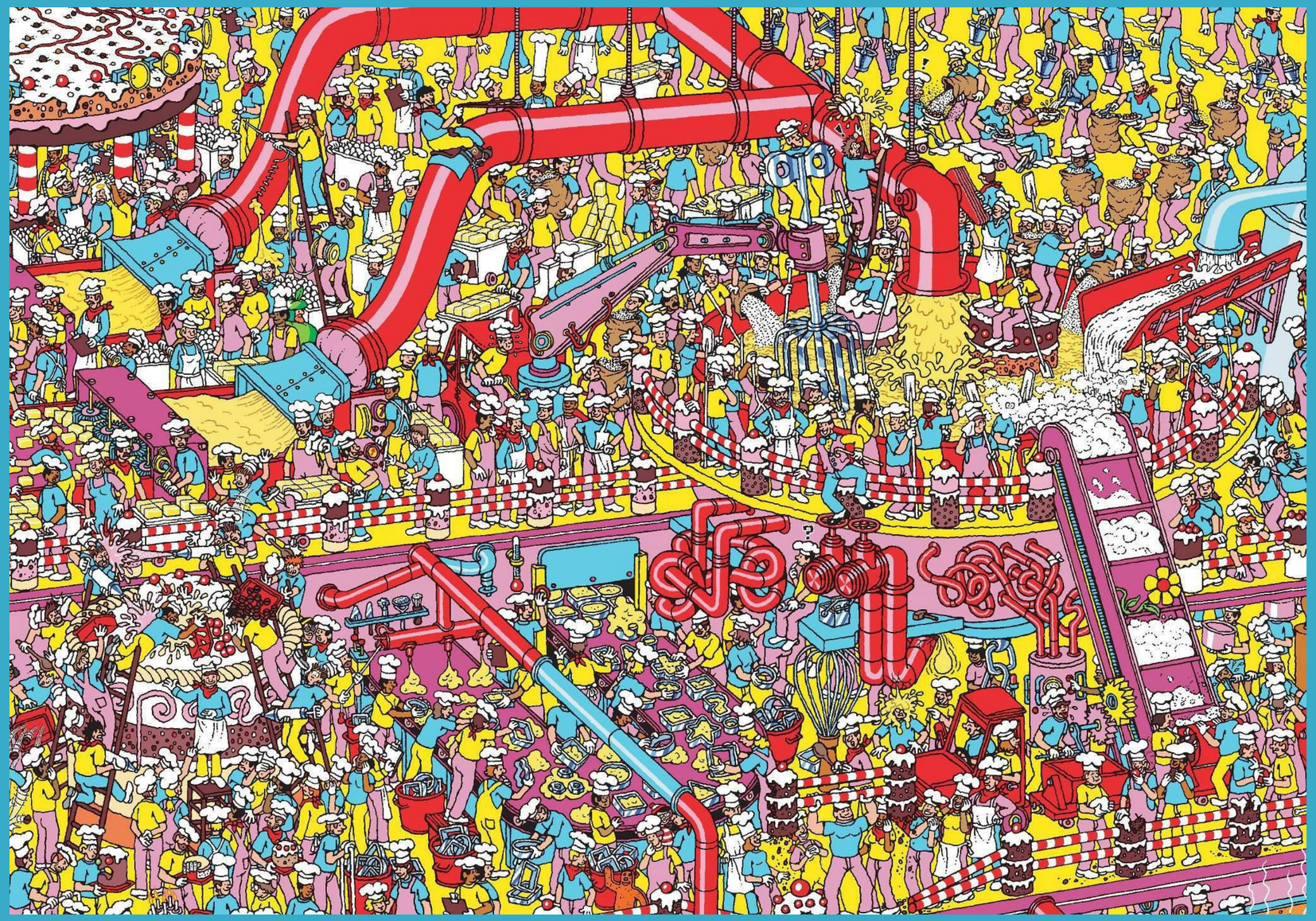
1990s and earlier	2000s	2010s
Coupling		
Pre-SOA (monolithic) Tight coupling	**Traditional SOA** Looser coupling	**Microservices** Decoupled

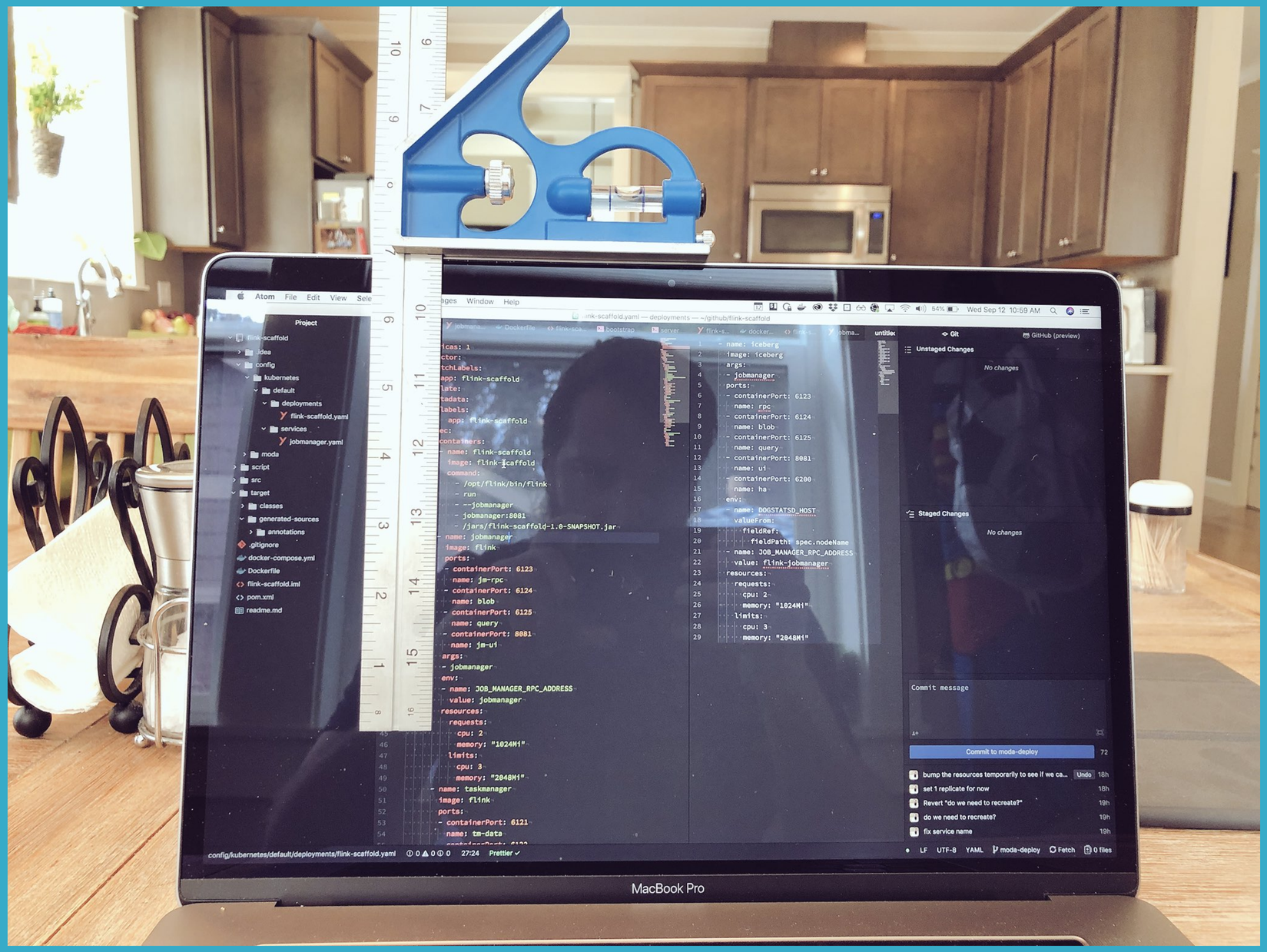
The Premise Of Microservices...

...And What Happens

Tools Of The Trade

Where's My Service?



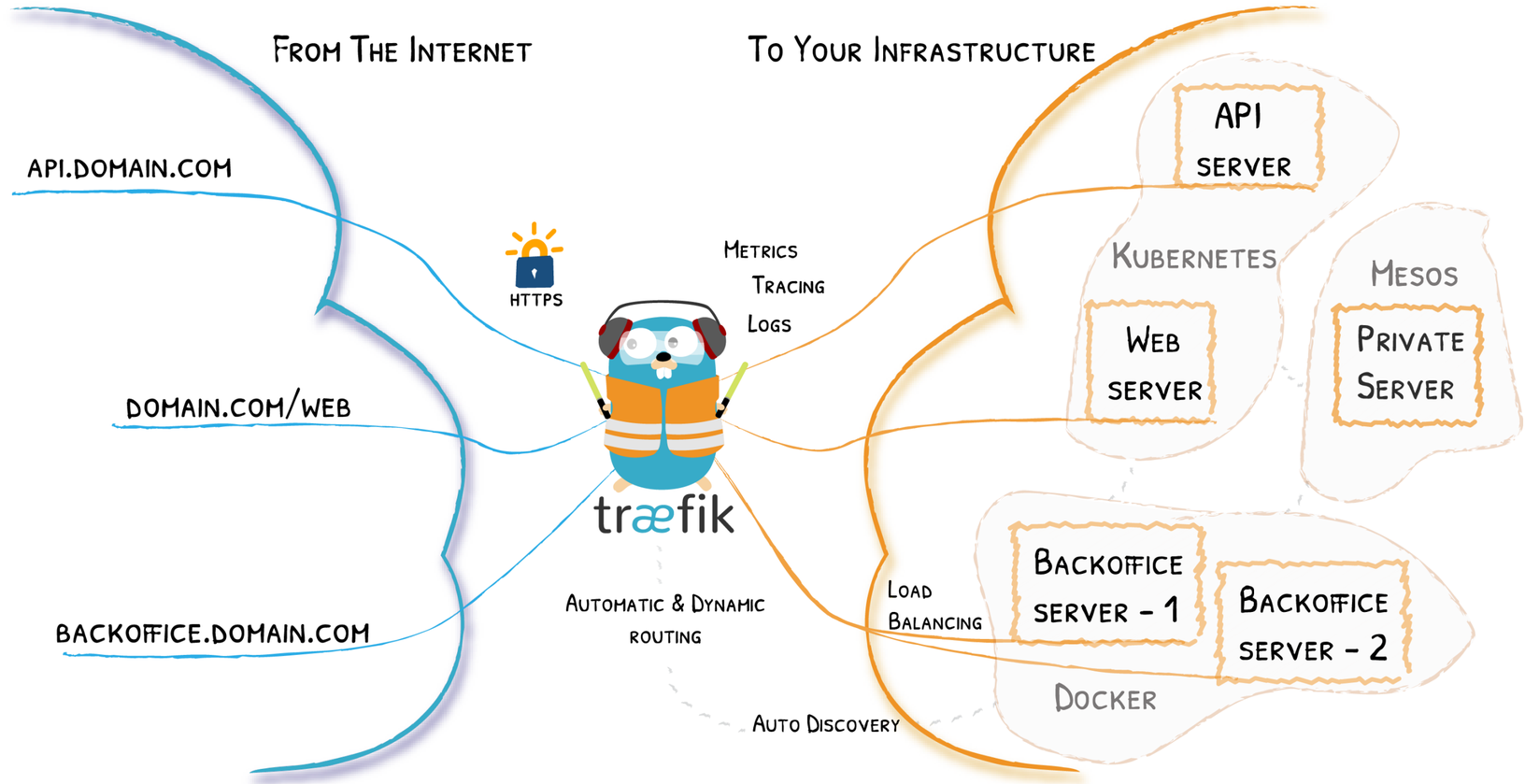


Source: https://twitter.com/Caged/status/1039937162769096704

What If I Told You?

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

Here Comes Traefik!

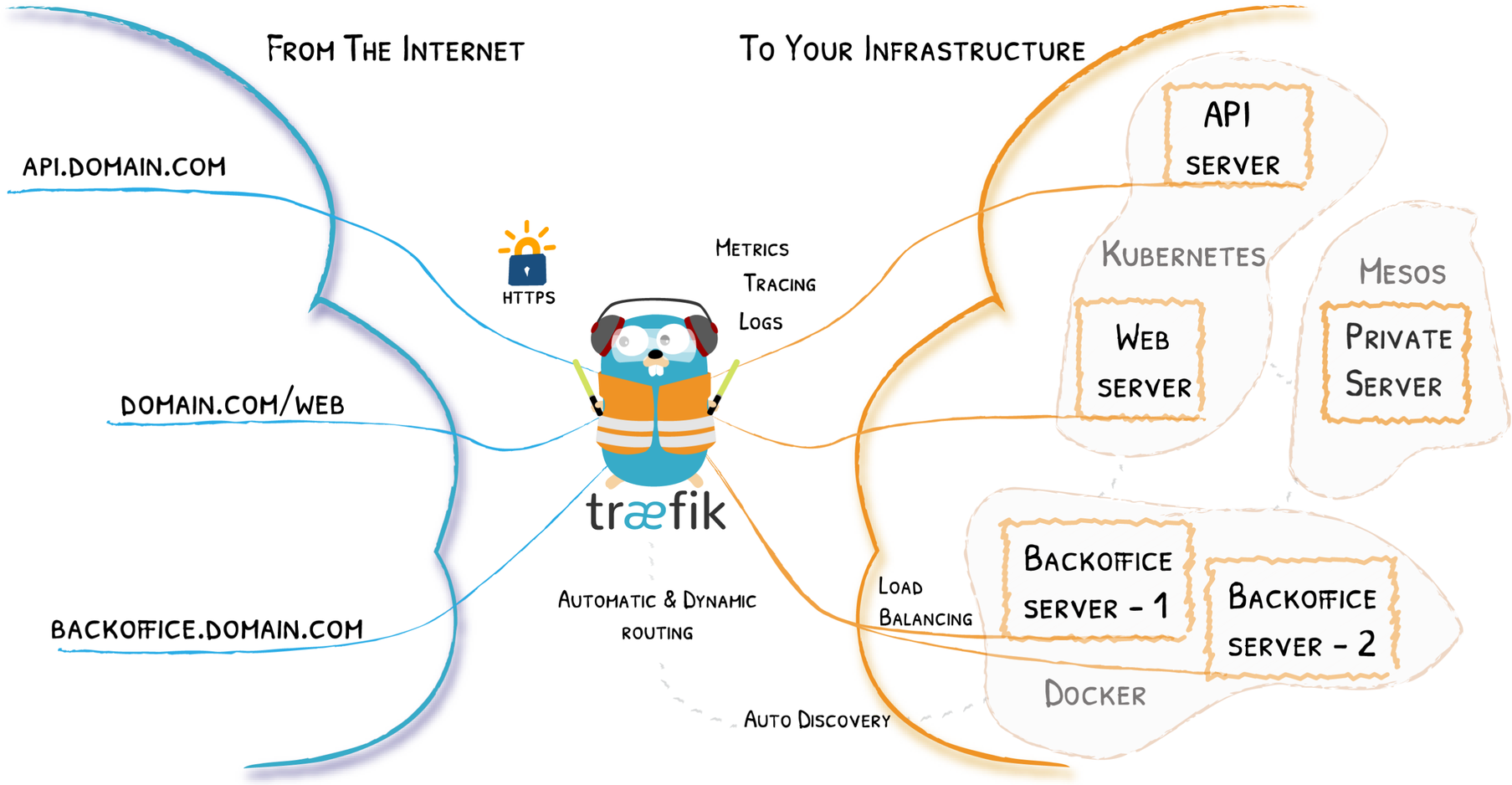


Traefik Project

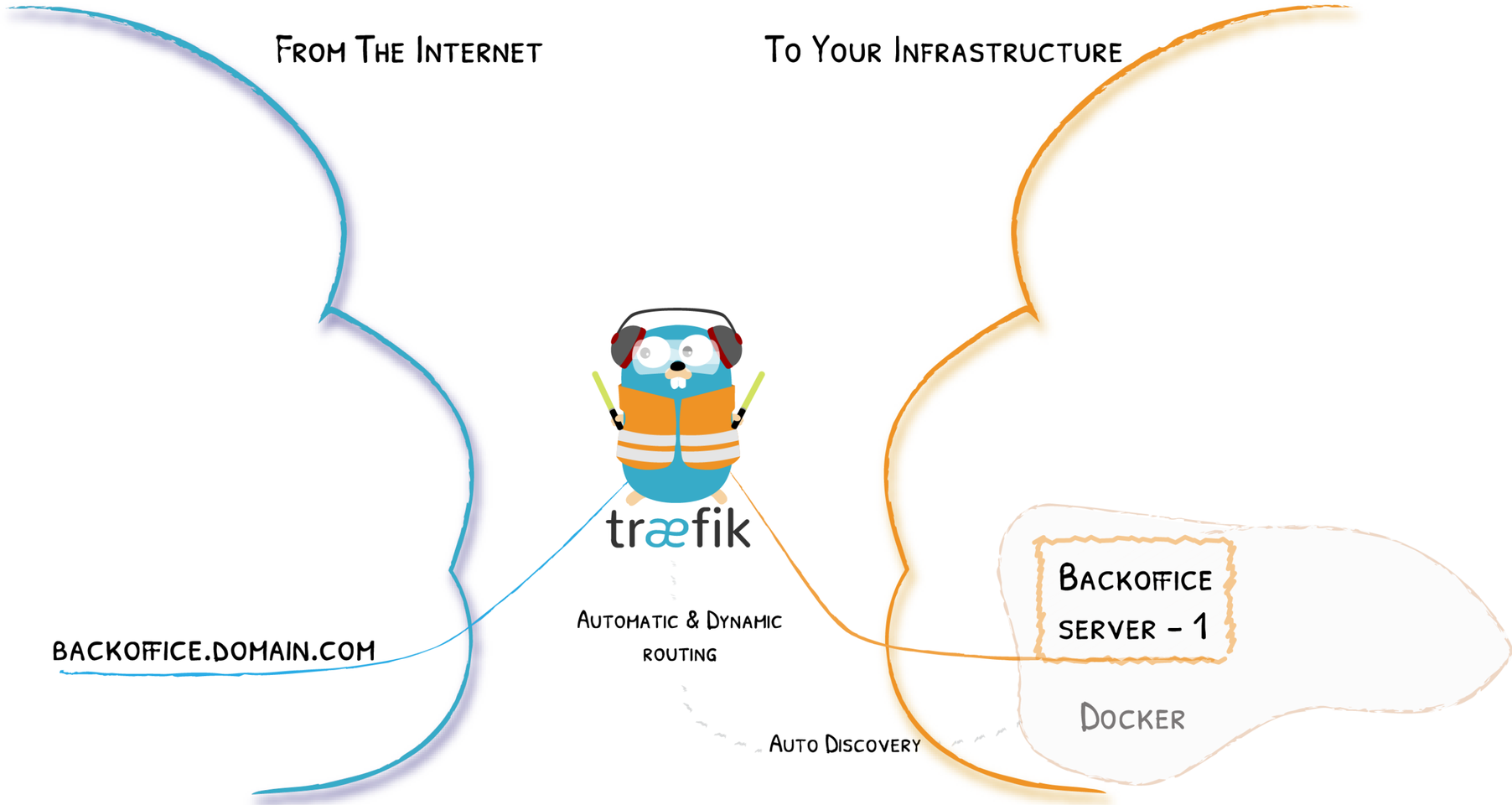
- https://github.com/containous/traefik
- MIT License
- Written in Go
- 21,000+ ☆
- 600M+ ⬇
- 350+ 👷

Traefik Core Concepts

Remember The Diagram?



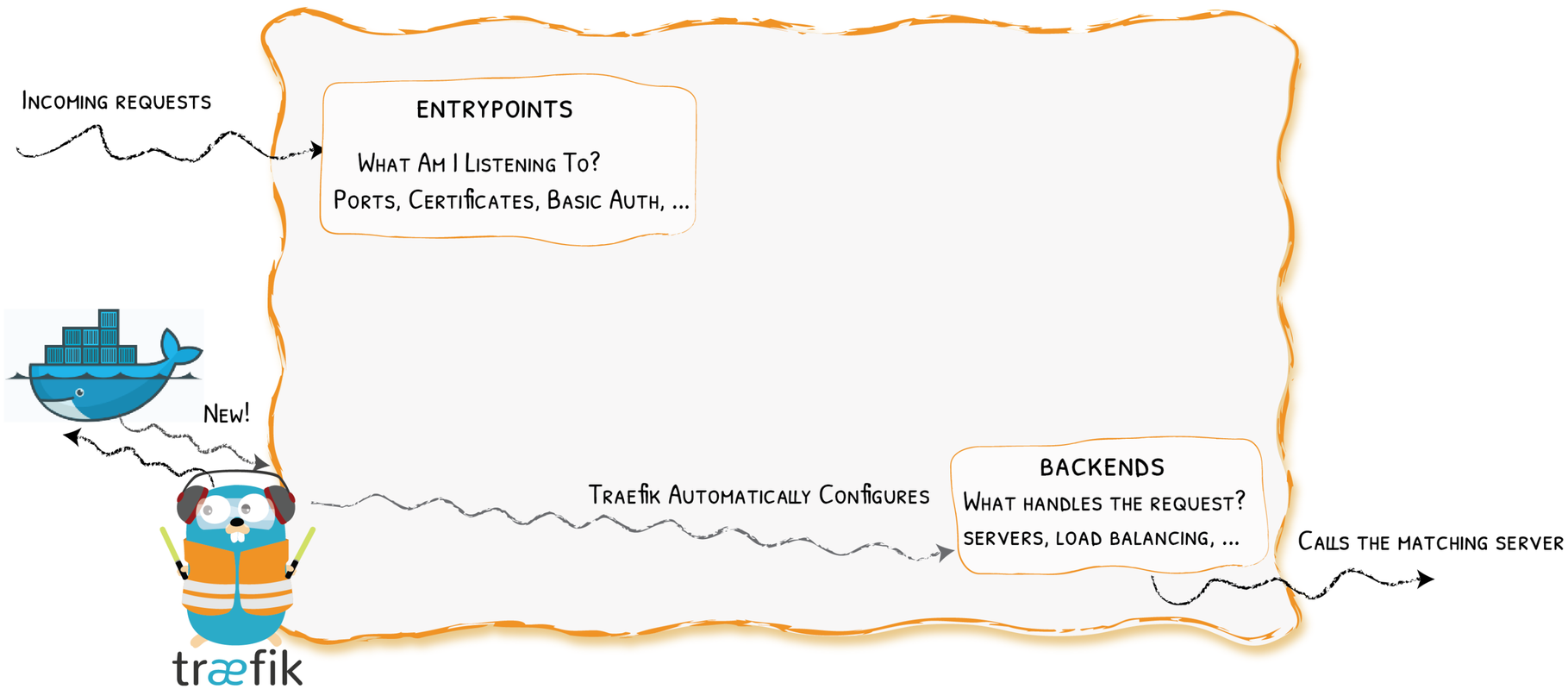
Let's Simplify



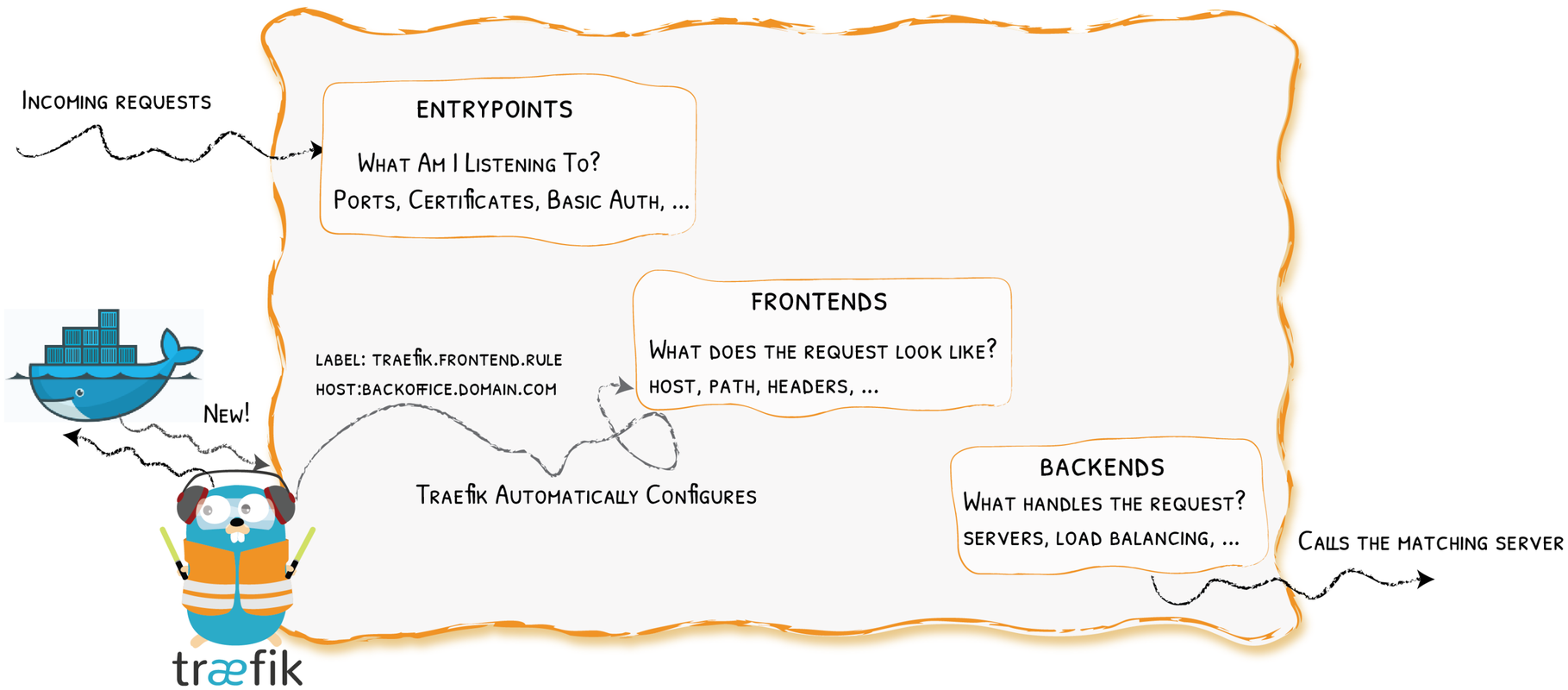
Providers

Entrypoints

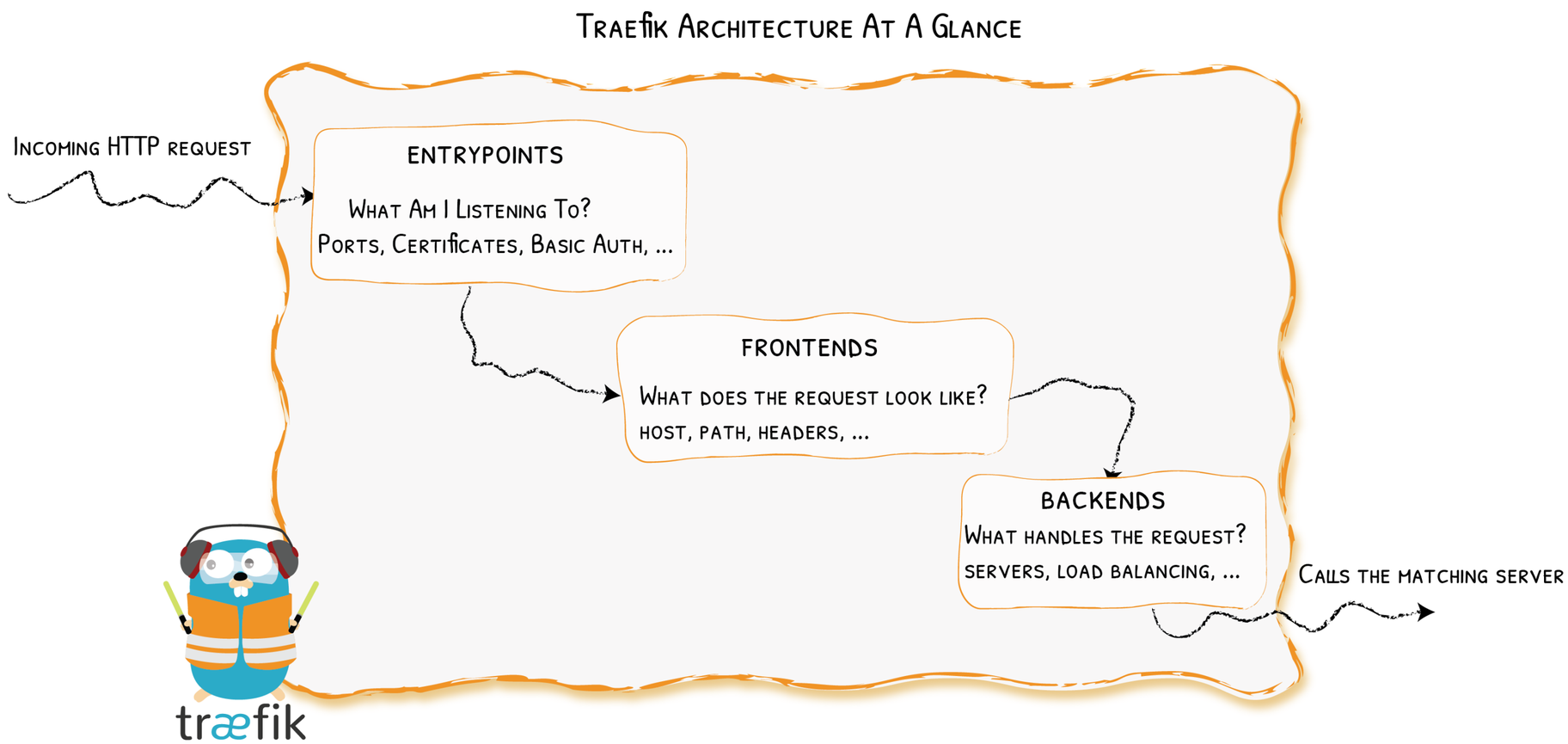
Backends


Incoming requests
ENTRYPOINTS
What Am I Listening To?
Ports, Certificates, Basic Auth, ...
New!
Traefik Automatically Configures
BACKENDS
What handles the request?
servers, load balancing, ...
Calls the matching server
træfik

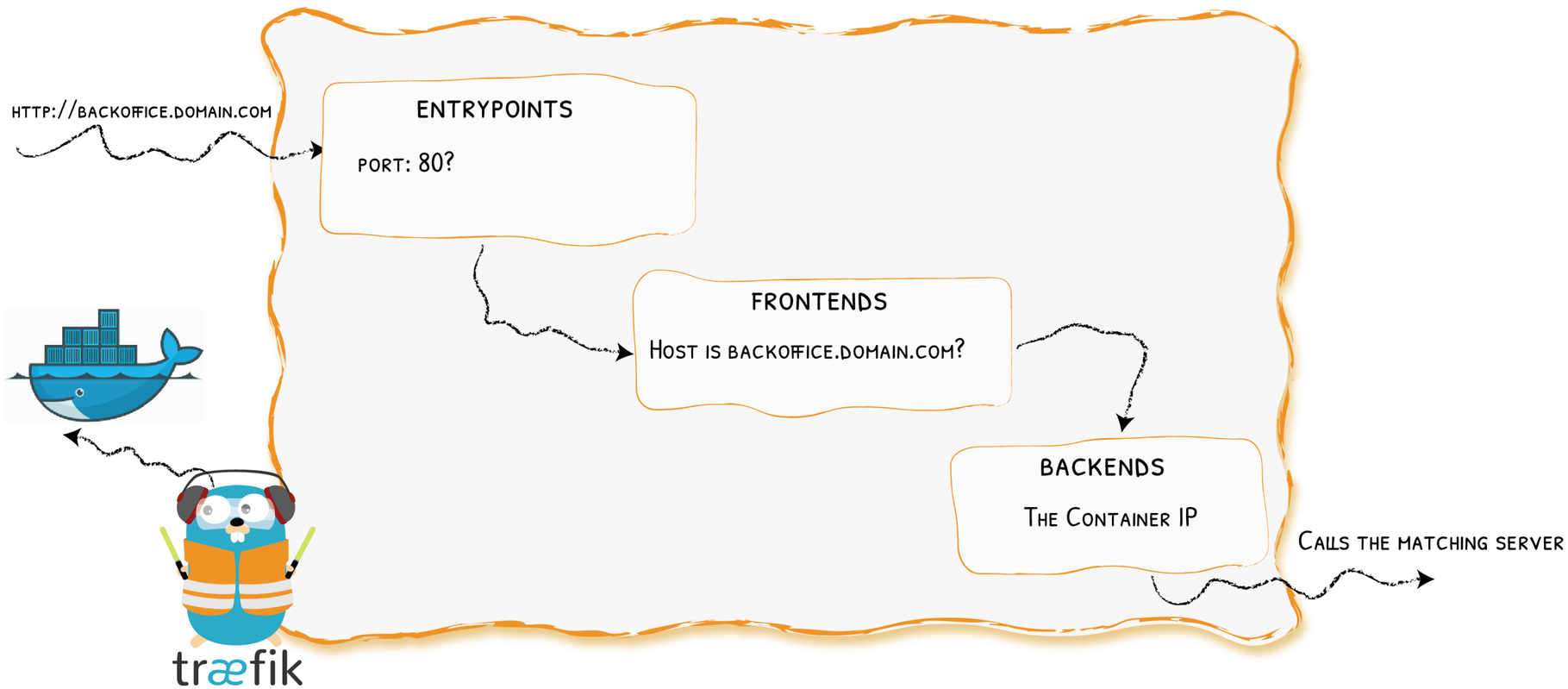
Frontends


Incoming requests
ENTRYPOINTS
What Am I Listening To?
Ports, Certificates, Basic Auth, ...
FRONTENDS
What does the request look like?
host, path, headers, ...
BACKENDS
What handles the request?
servers, load balancing, ...
Calls the matching server
label: traefik.frontend.rule
host:backoffice.domain.com
Traefik Automatically Configures
New!
træfik

At A Glance


Traefik Architecture At A Glance
Incoming HTTP request
ENTRYPOINTS
What Am I Listening To?
Ports, Certificates, Basic Auth, ...
FRONTENDS
What does the request look like?
host, path, headers, ...
BACKENDS
What handles the request?
servers, load balancing, ...
Calls the matching server
træfik

In Practice


HTTP://BACKOFFICE.DOMAIN.COM
ENTRYPOINTS
PORT: 80?
FRONTENDS
HOST IS BACKOFFICE.DOMAIN.COM?
BACKENDS
THE CONTAINER IP
CALLS THE MATCHING SERVER
træfik

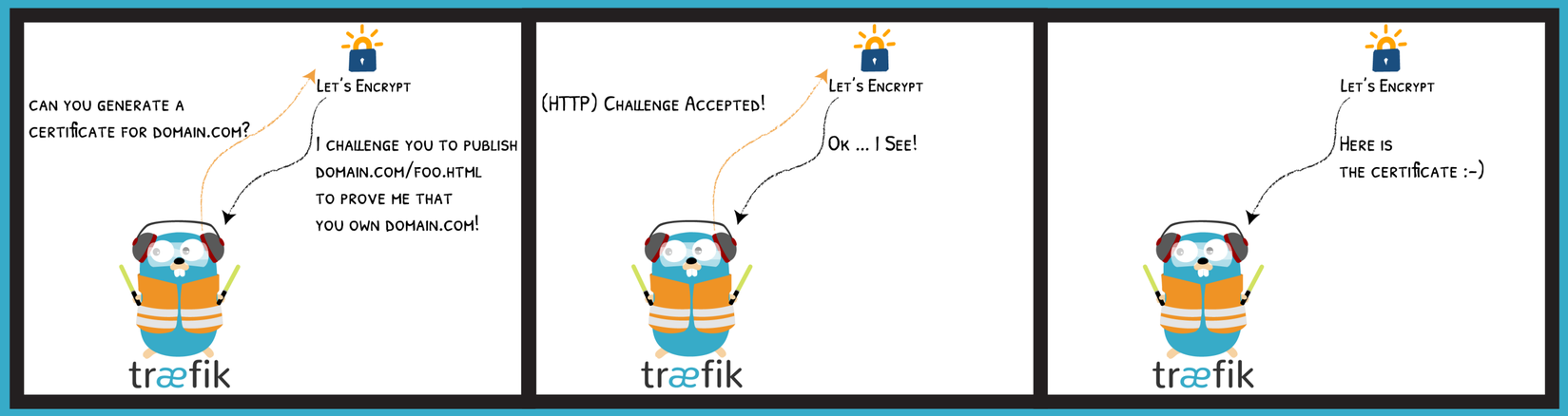
Show Me The Configuration!

Keep It Simple

- With 🐳:

```
entrypoint:
  image: traefik:v1.7
  command:
    - "--docker"
    - "--docker.domain=mycompany.org"
    - "--acme.email=ssl-admin@mycompany.org"
    - "--acme.httpChallenge.entryPoint=http"
    # Or you could use a TOML file with "--configFile=/etc/traefik/traefik.toml
  volumes:
    - /var/run/docker.sock:/var/run/docker.sock
```

HTTPS For Everyone With Let's Encrypt



- TLS, DNS and HTTP challenges supported

With : Simple Backend

```
# https://www.mycompany.org -> http://webserver:80/
webserver:
    image: nginx:alpine
    labels:
      - "traefik.frontend.rule=Host:www.mycompany.org"
```

With : Context

```
# https://mycompany.org/jenkins -> http://jenkins:8080/jenkins
jenkins:
    image: jenkins/jenkins:lts
    labels:
      - "traefik.frontend.rule=PathPrefix:/jenkins"
      - "traefik.port=8080" # Because 50000 is also exposed
    environment:
      - JENKINS_OPTS=--prefix=/jenkins
```

With : Rewrites

```
# https://mycompany.org/gitserver -> http://gitserver:3000/
gitserver:
    image: gitea/gitea:1.5
    labels:
      - "traefik.frontend.rule=PathPrefixStrip:/gitserver"
      - "traefik.port=3000" # Because 22 is also exposed
```

With  : Websockets

```
# https://mycompany.org/webterminal -> http://webterminal:7681/
webterminal:
    image: tsl0922/ttyd
    labels:
      - "traefik.frontend.rule=PathPrefixStrip:/webterminal"
    expose:
      - "7681"
```

Traefik With ⎈

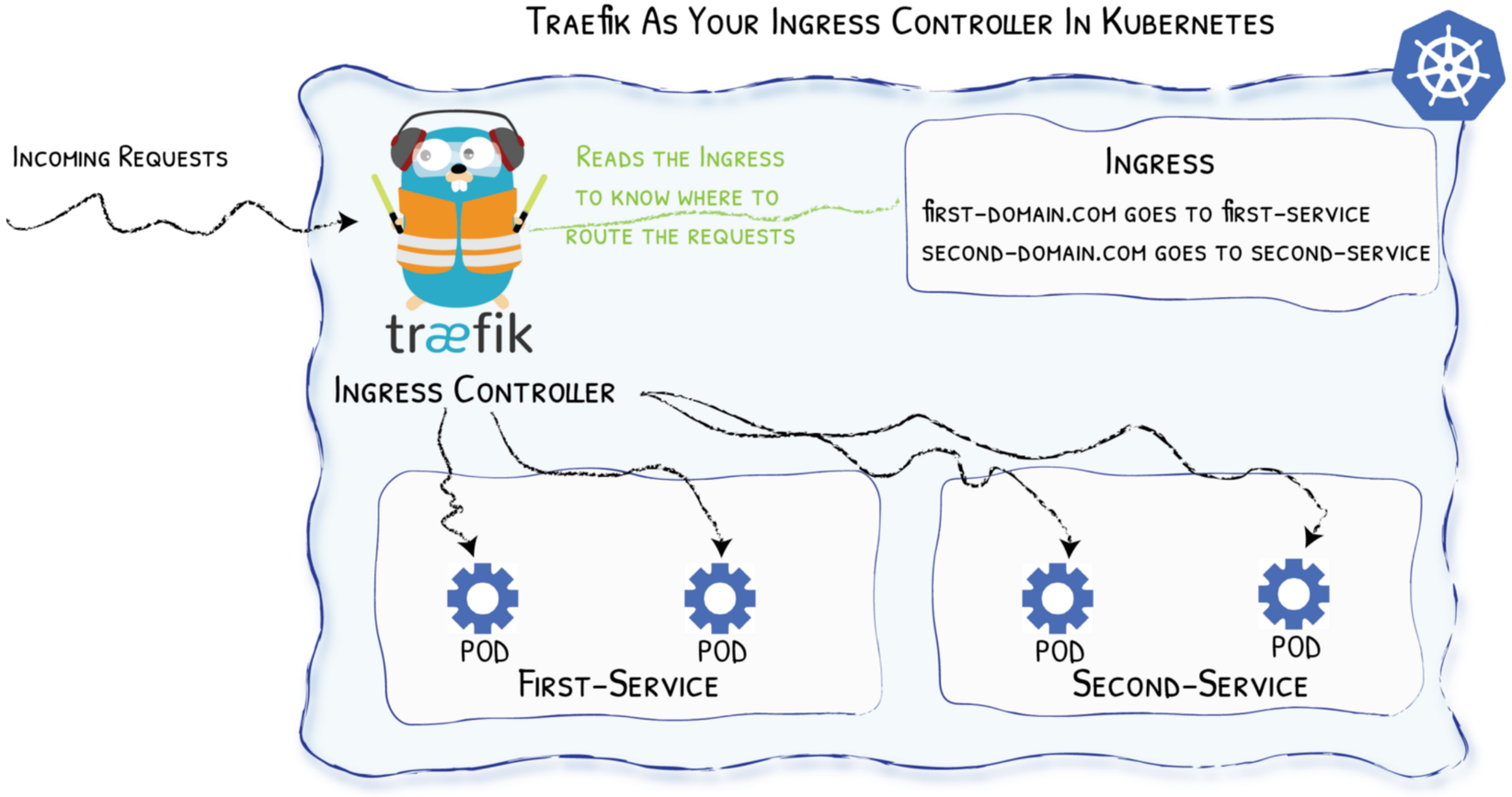


Diagram from https://medium.com/@geraldcroes

Did You Say YAML?

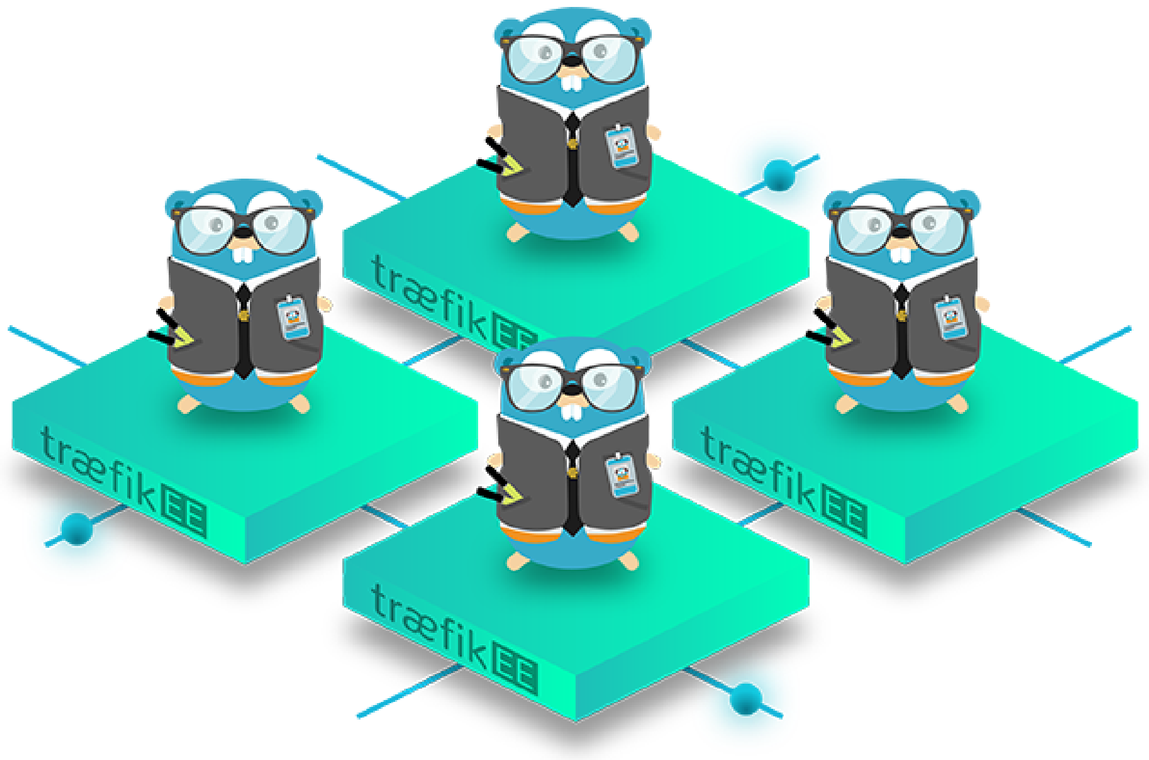
```yaml
apiVersion: extensions/v1beta1
kind: Ingress
metadata:
  annotations:
    # kubernetes.io/ingress.class: 'nginx'
    kubernetes.io/ingress.class: 'traefik'
spec:
  rules:
  - host: mycompany.org
    http:
      paths:
      - path: "/whoami"
        backend:
          serviceName: whoami
   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

We Missed Talking About...

The Herd

You came to the wrong neighbour

Traefik Comes In Herd



HIGH AVAILABILITY

træfik ENTERPRISE EDITION

SECURITY

træfik ENTERPRISE EDITION

SCALABILITY

træfik ENTERPRISE EDITION

NTERNET

TO YOUR INFRA

Control Plane

traefikEE

?

ORCHESTR

Data Plane

traefikEE

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

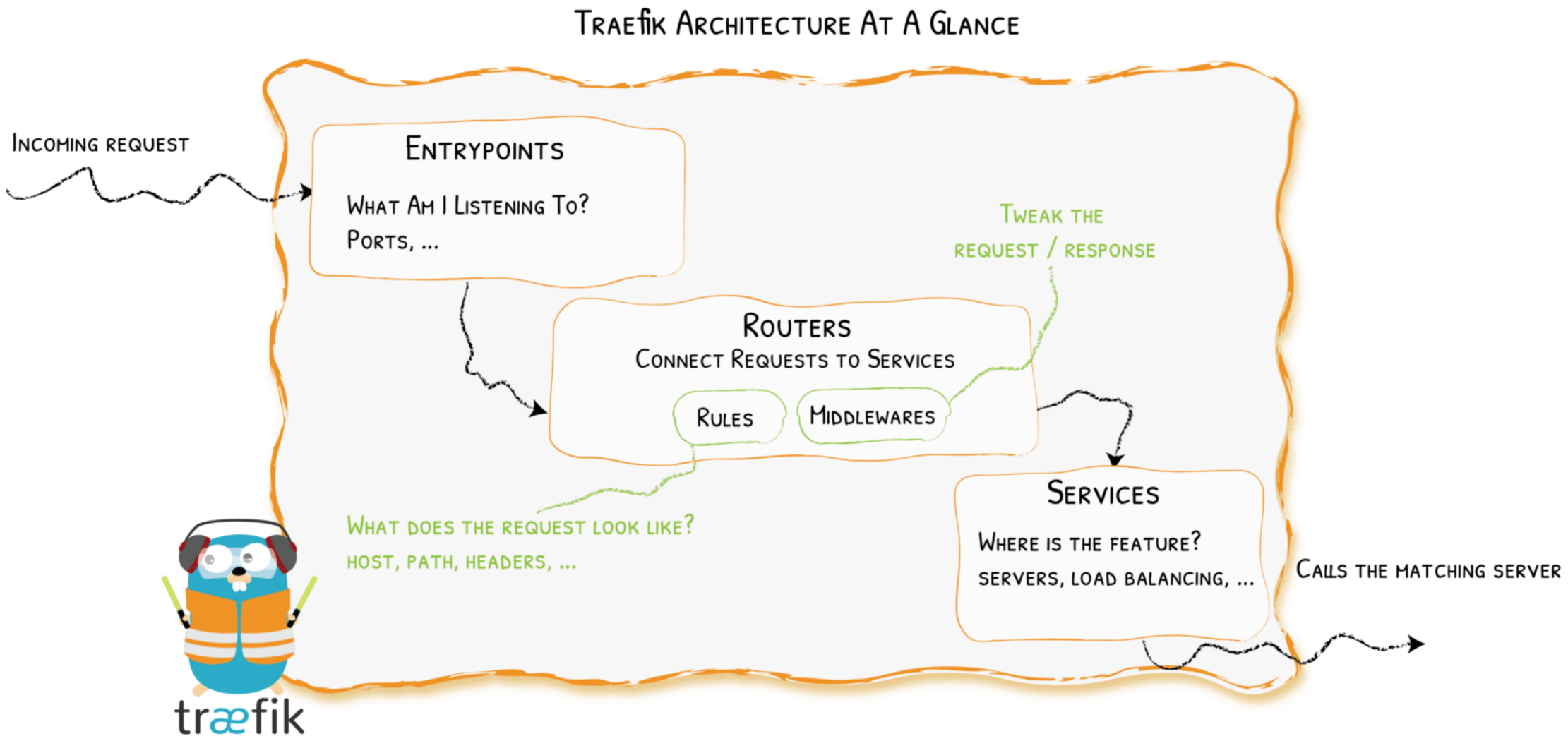
But

What About Open Source?

BACK TO TRAEFIK PART 2.0

Revamped Documentation

Clarified Concepts

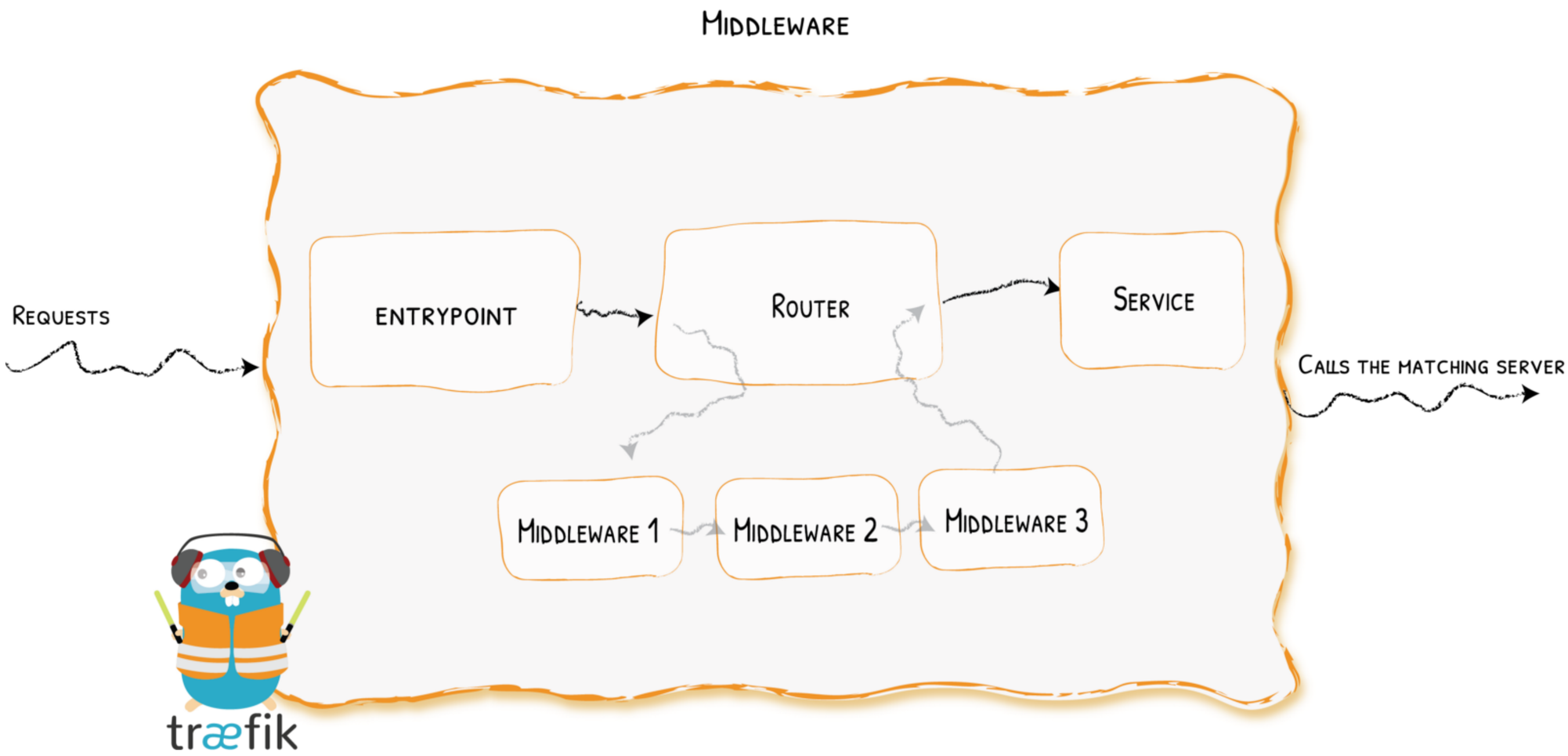


Expressive Routing Rule Syntax


```
# Send both requests to backend service:
#   https://api.mycompany.com/v2
#   https://api-v2.mycompany.com

rule=(Host('api.mycompany.com') && PathPrefix('/v2')) || Host('api-v2.mycompany.com')
```

Middlewares



⎈ CRD - Custom Resources Definition

```yaml
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
```

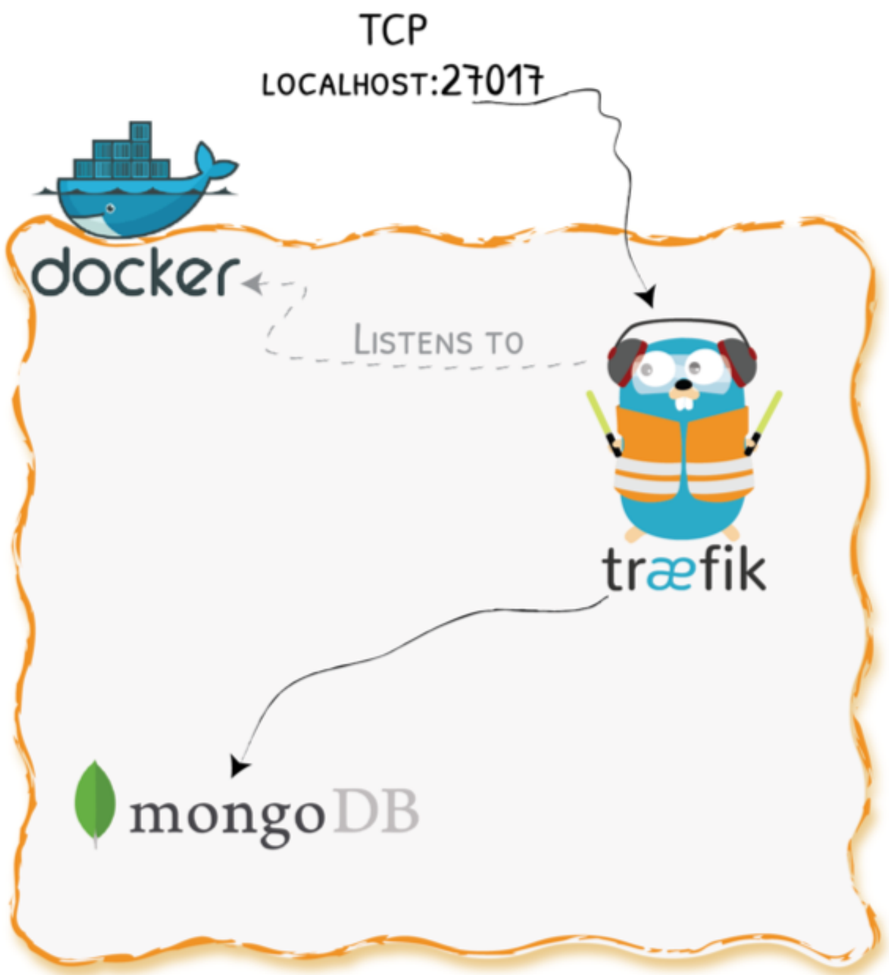
traefik

HTTP

&

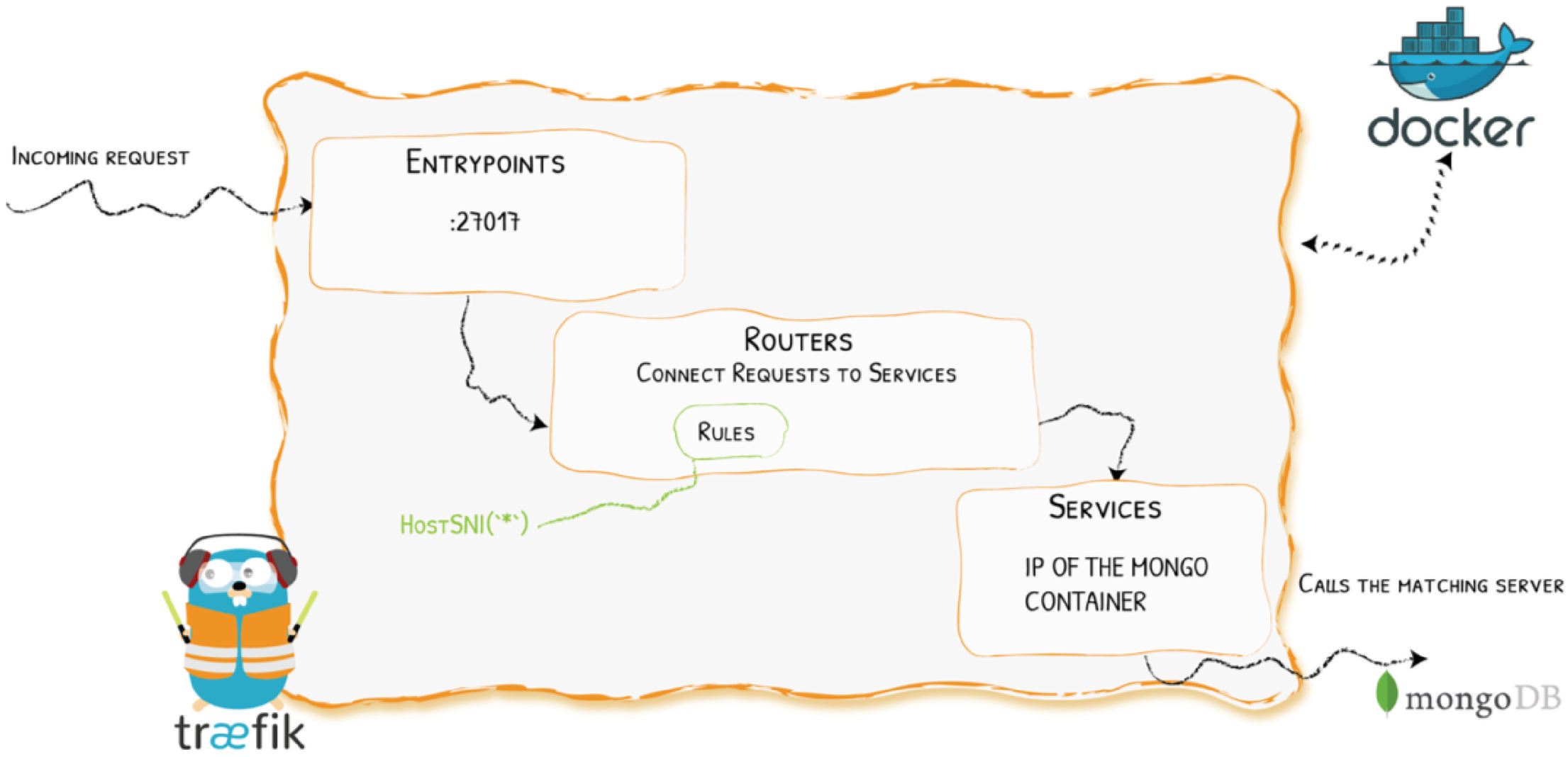
TCP

Demo 1 - Straightforward TCP Routing

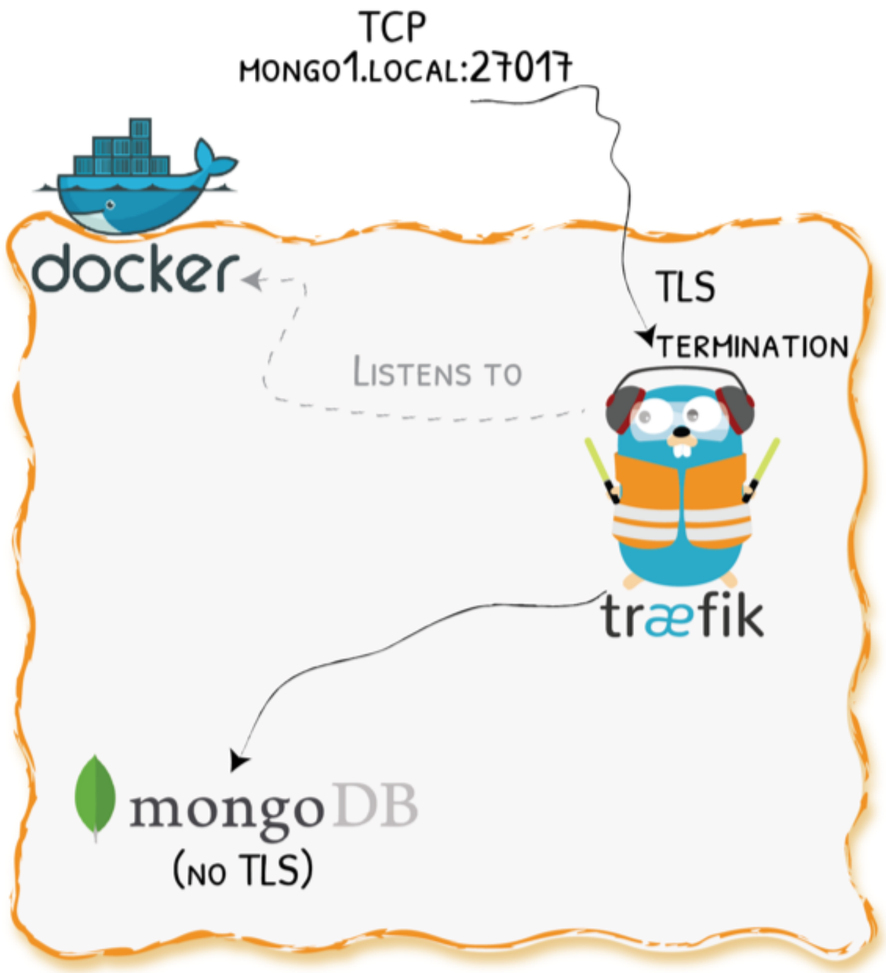


Demo Code on

Demo 1 - Configuration

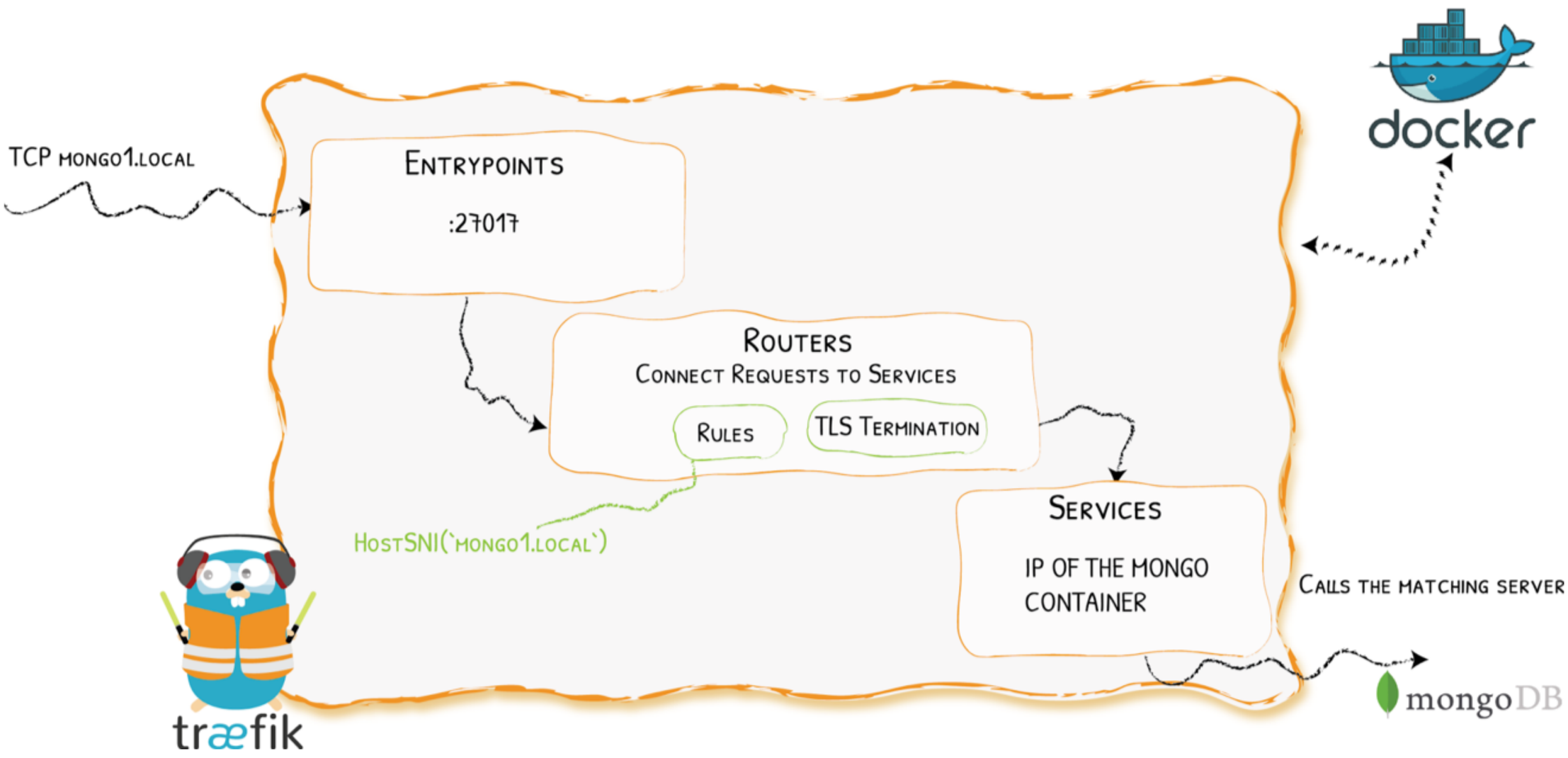


Demo 2 - Let's Add TLS To TCP With Traefik

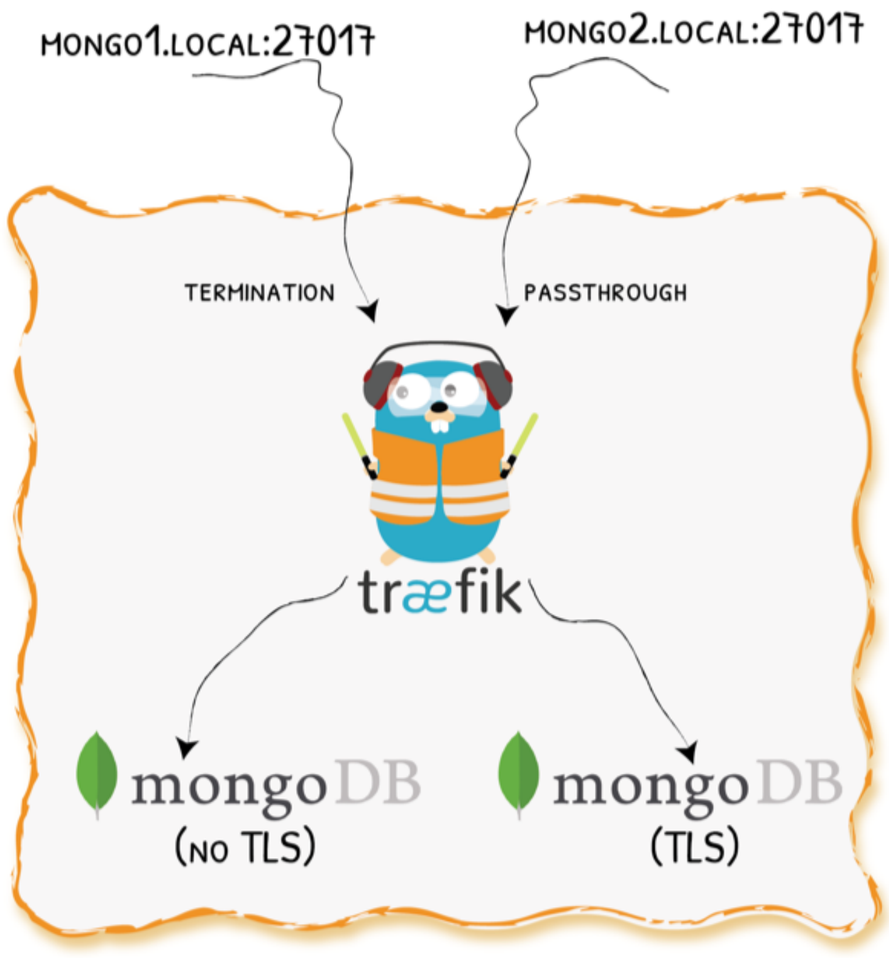


Demo Code on

Demo 2 - Configuration

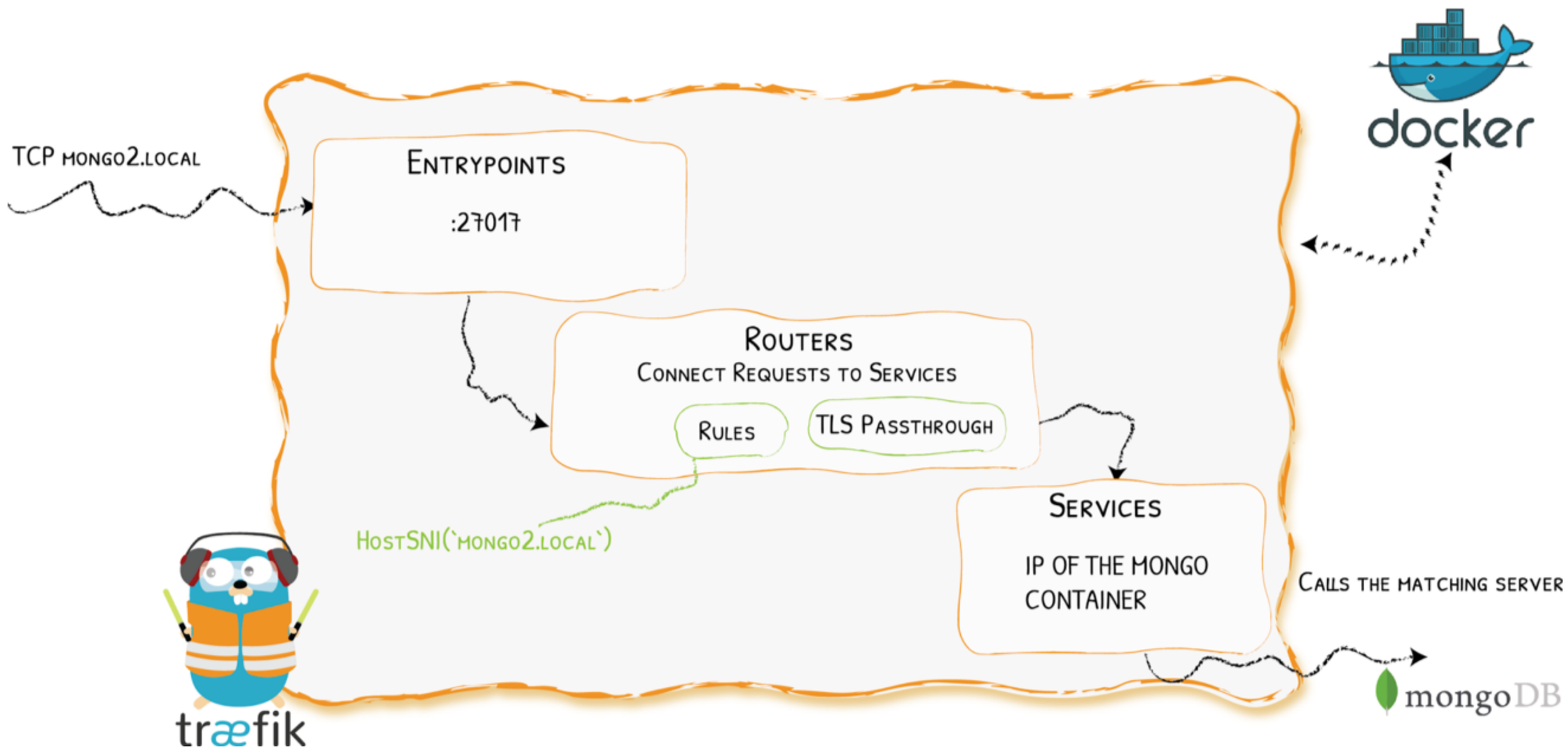


Demo 3 - SNI Routing + TLS Passthrough

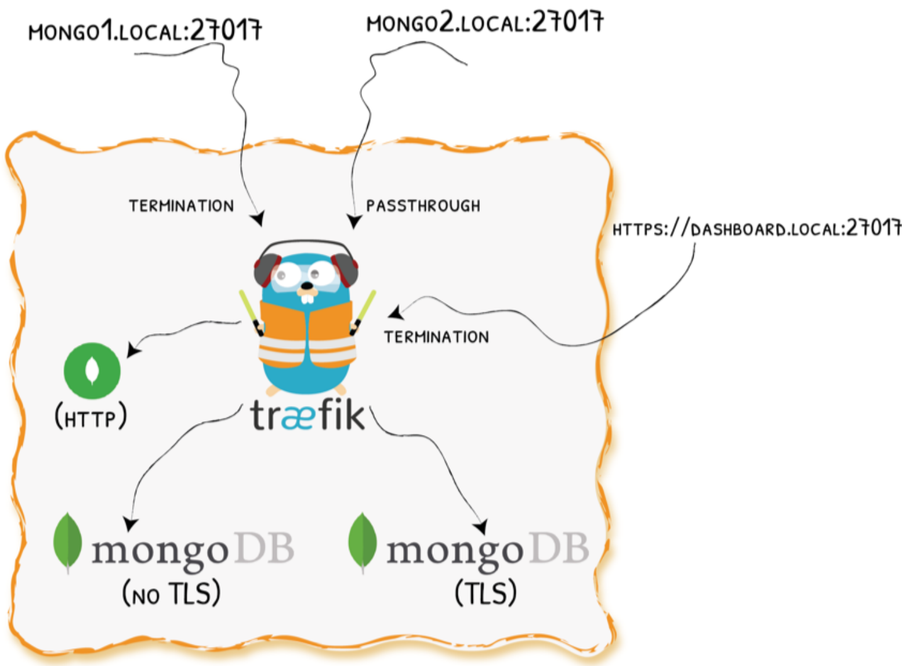


Demo Code on

Demo 3 - Configuration



Demo 4 - Muxing HTTPS And TCP On The Same Port



Demo Code on

More To Come

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

More Info

bit.ly/traefik-v2

TO BE

CONTINUED...

We Have Stickers!

Thank You!

@DamienDuportal

dduportal

- Slides (HTML): https://containous.github.io/slides/bbl-decathlon-2019
- Slides (PDF): https://containous.github.io/slides/bbl-decathlon-2019/slides.pdf
- Source on : https://github.com/containous/slides/tree/bbl-decathlon-2019