

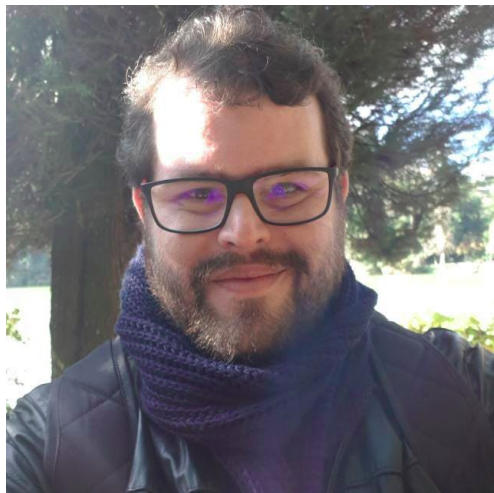


Træfik

da PoC a Prod

@marcopollivier

Marco Ollivier - @marcopollivier



- Analista de Sistemas @ Infnet



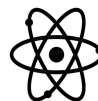
- Software Engineer @ OLX



- ❤️ Go, Java, Docker e Linux

- Segundo o Datafolha, 7 anos de experiência

- Entusiasta de Astronomia e Física



Agenda

- Um pouco de teoria
 - O que é um Proxy Reverso?
 - O que é um Balanceador de Carga?
 - Por que mudar?
 - Breve resumo das soluções do mercado
 - Apresentando o Træfik
 - Funcionalidades
 - Instalação
 - da Poc - Prova de Conceito
 - a Prod - Casos de sucesso
 - Resumo
 - Agradecimentos
 - Perguntas
-

—

Antes de qualquer
coisa, um pouco de
teoria.

Micro Serviços

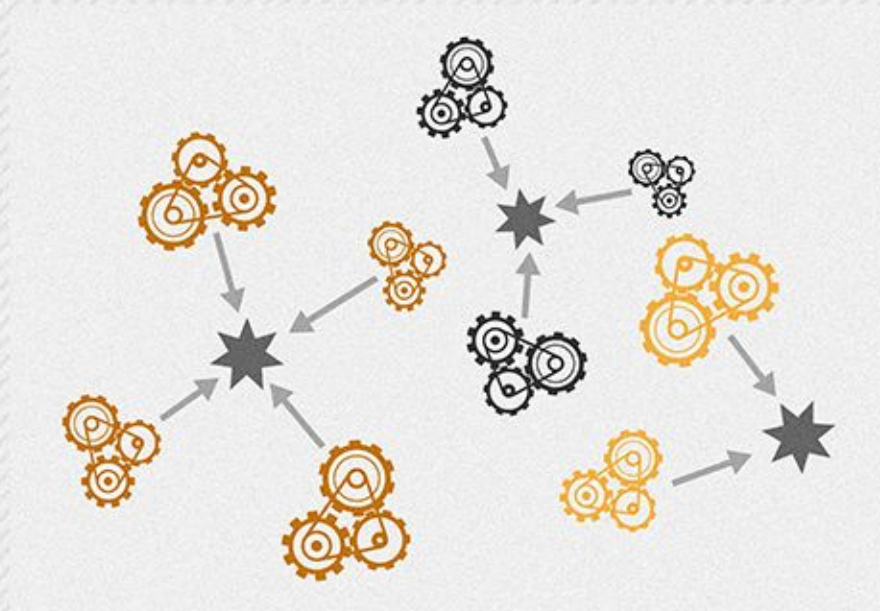
- Desacoplamento
- Escalabilidade
- Análise e investigação mais fácil e rápida
- Maior facilidade na Entrega Contínua

Porém...

- Gerenciar fica um pouco mais complicado
-



Monolithic



Microservices

Reverse Proxy

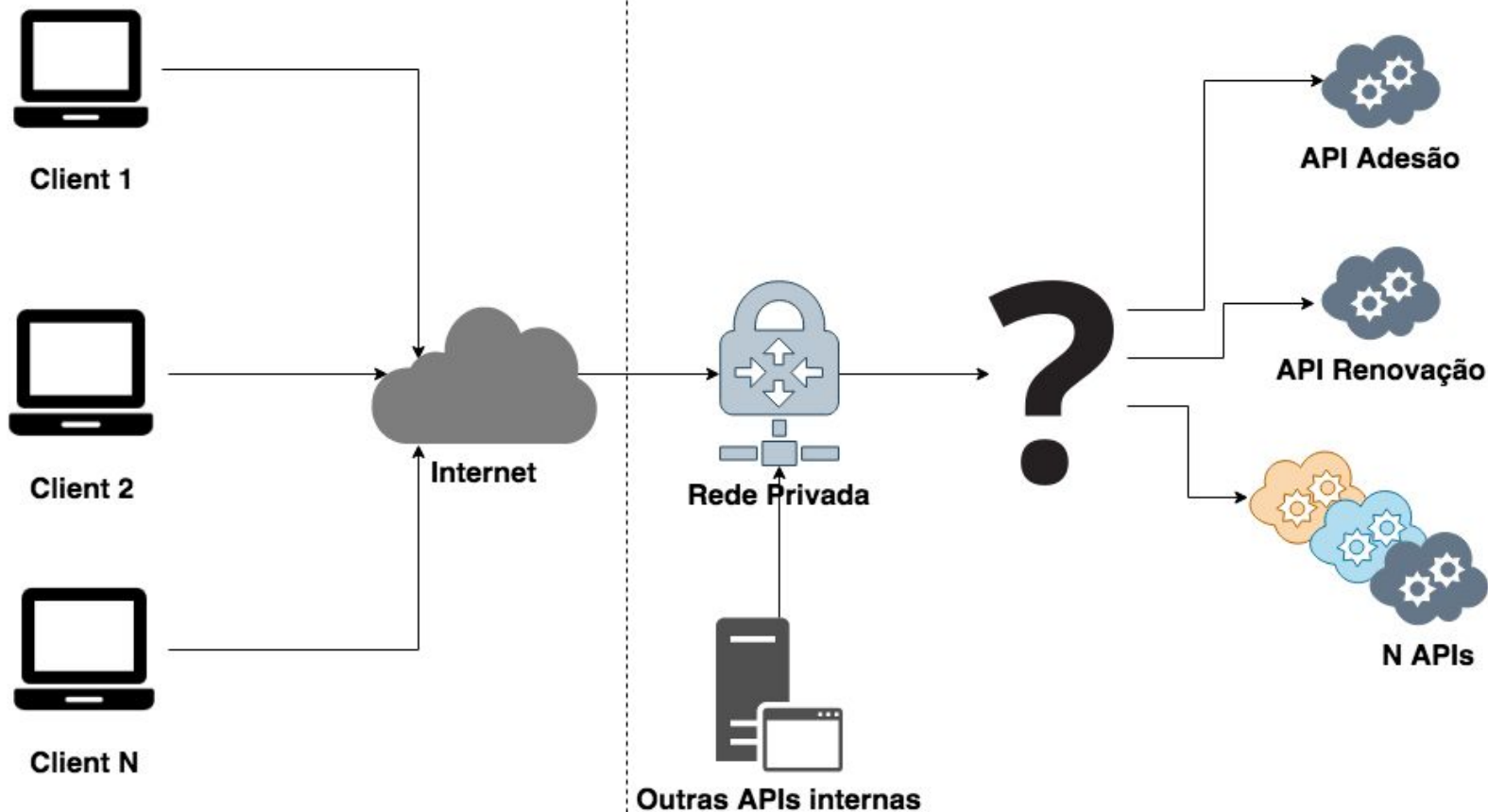
Cenário...

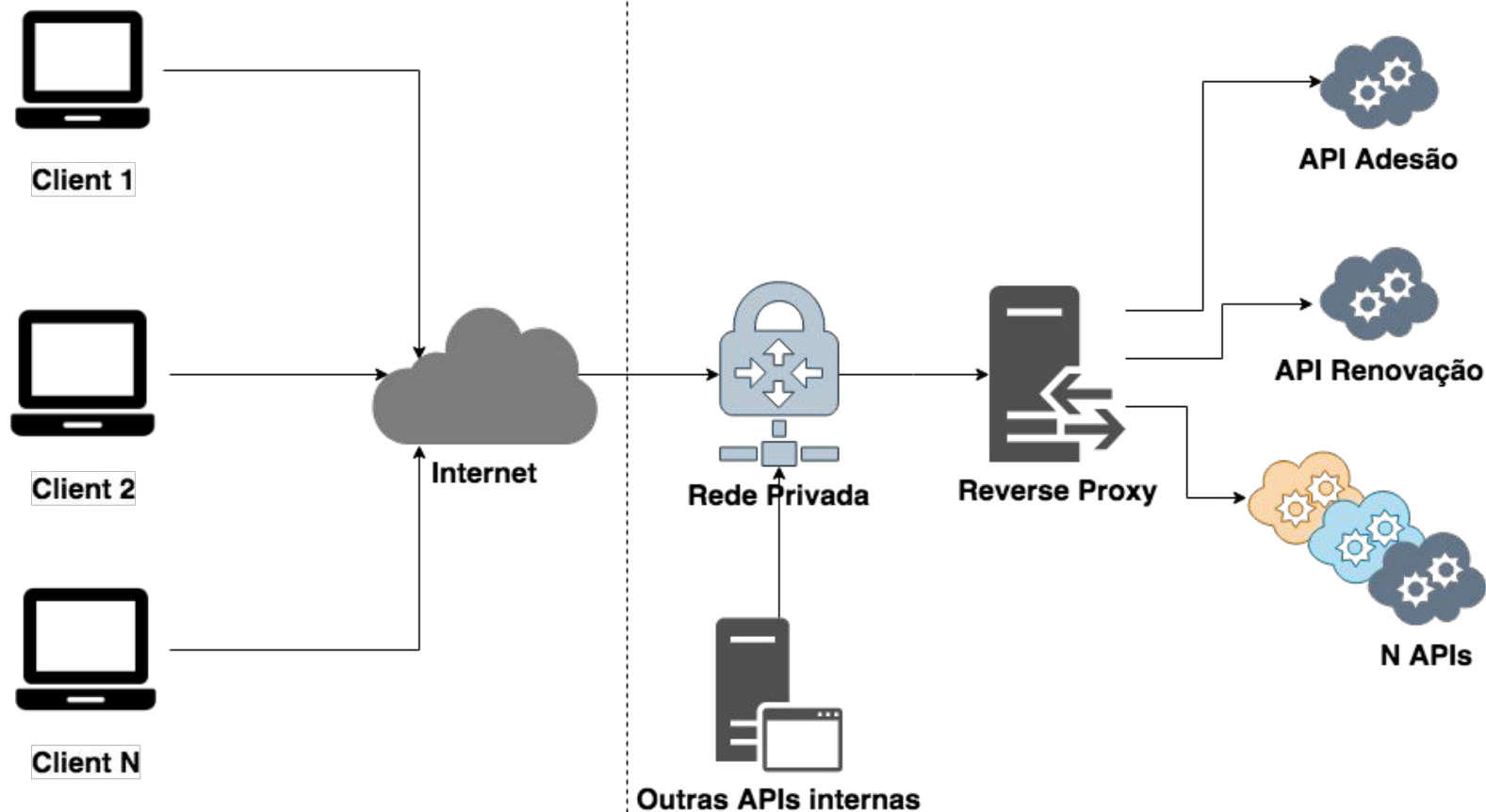
- Grupo com múltiplos microsserviços

 - Acesso externo ao seu contexto de solução:
 - Seja Internet
 - Serviços de terceiros
-

Reverse Proxy

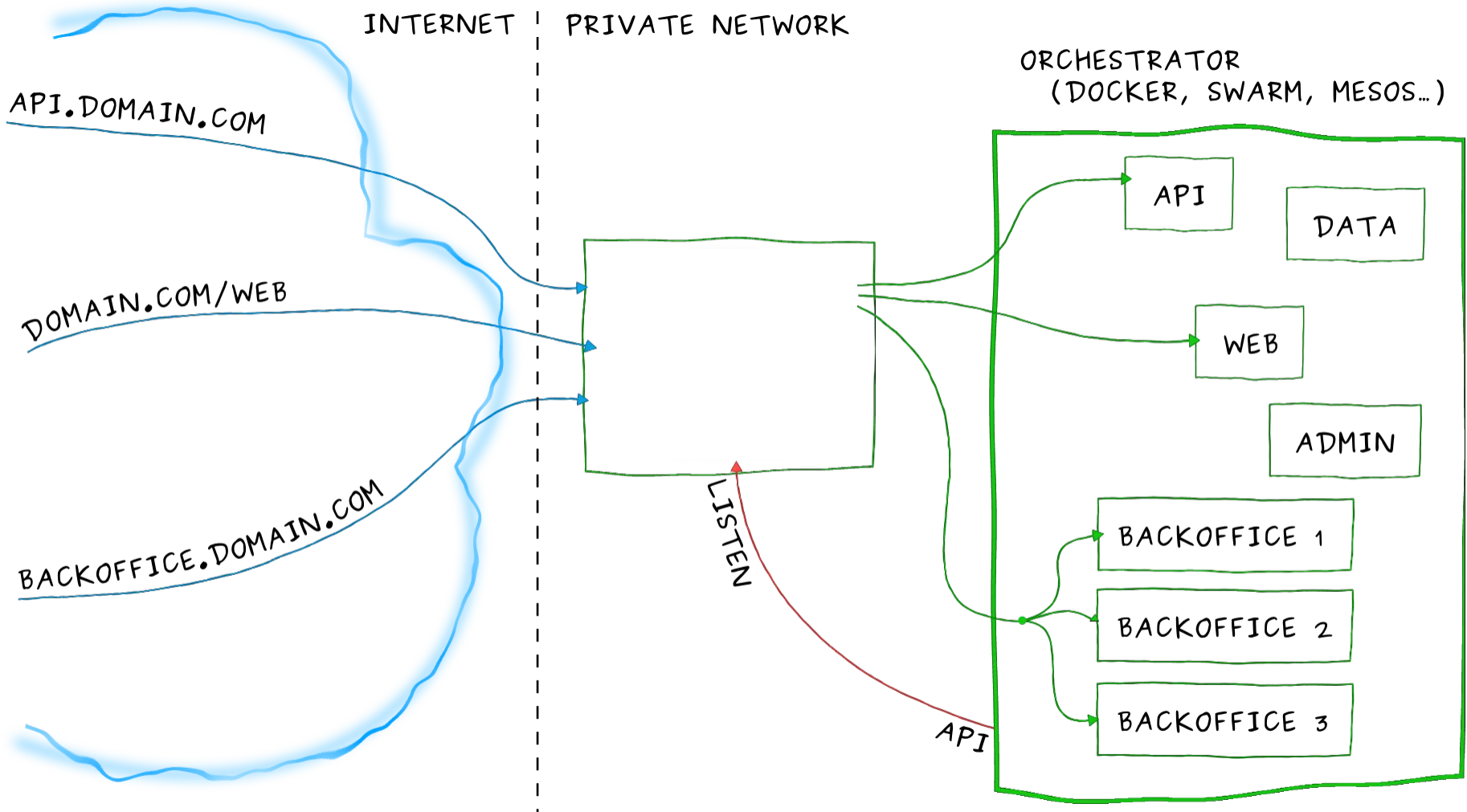
- > Como saber o que vai para onde?
 - > Como redirecionar as requisições?
-







SPOILER ALERT



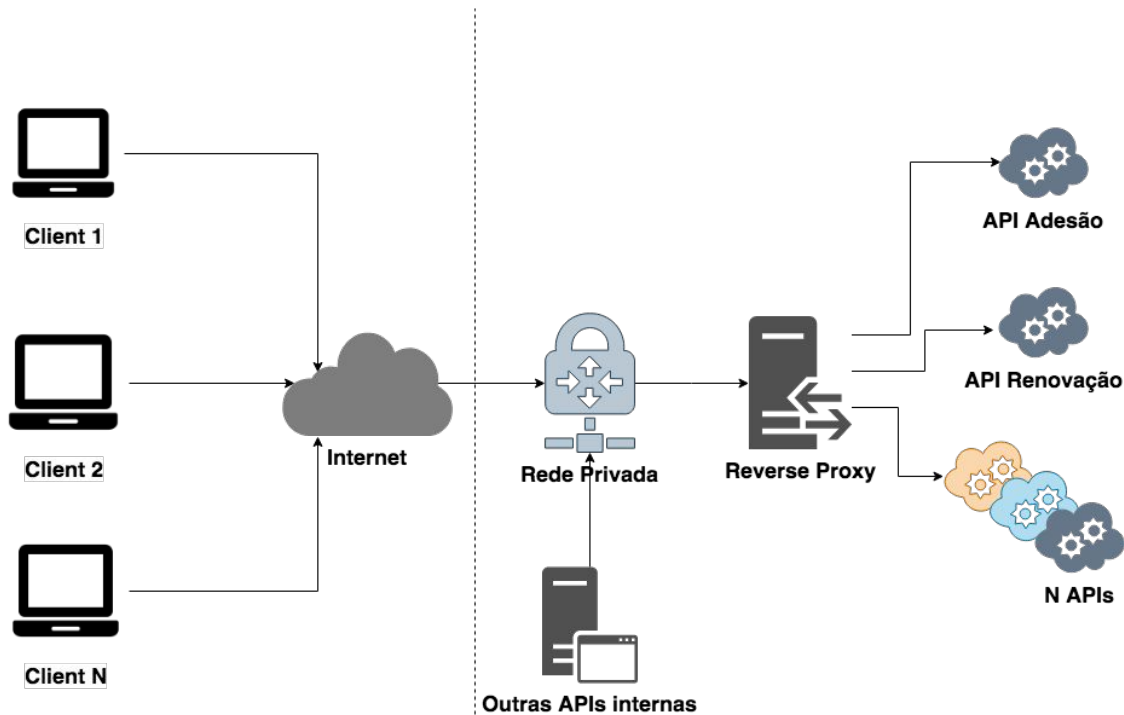
Reverse Proxy

- api.domain.com vai acessar as APIs via internet
 - domain.com/loja vai acessar um website via internet
 - backoffice.domain.com vai acessar uma api de backoffice via internet
 - api.private-domain vai acessar a mesma API de dentro da rede interna
-

Load Balancer

Retornando ao problema...

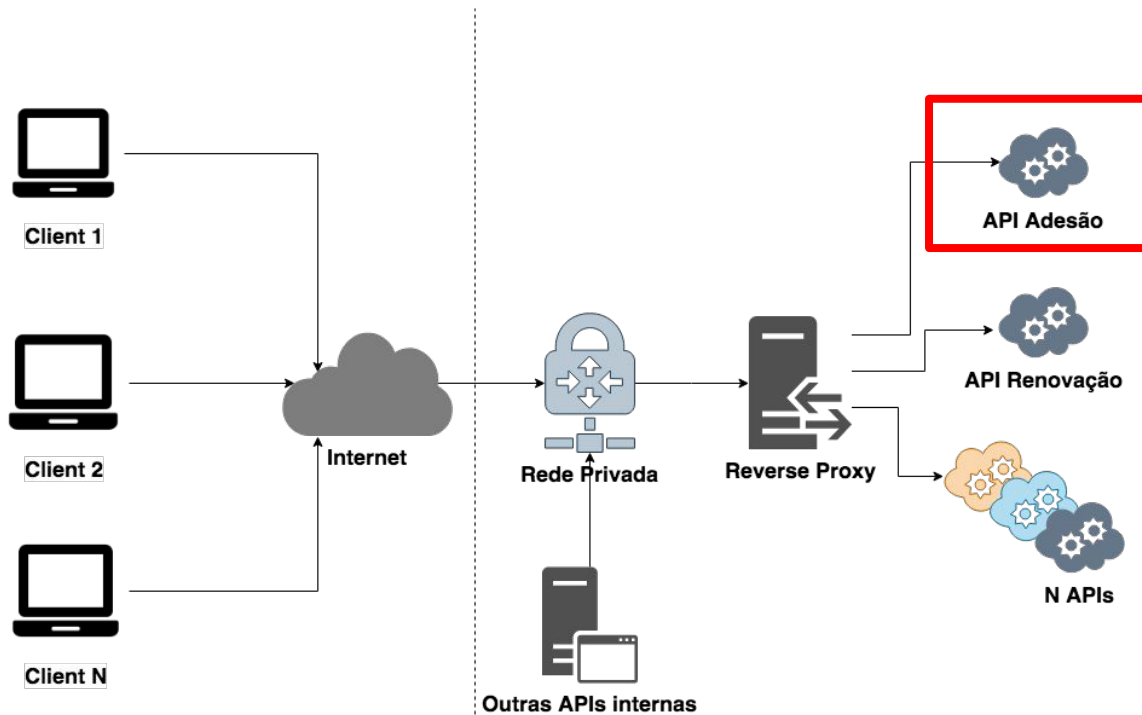
O que aconteceria se na nossa **API de Adesão**, só tivéssemos um servidor atendendo as requisições?



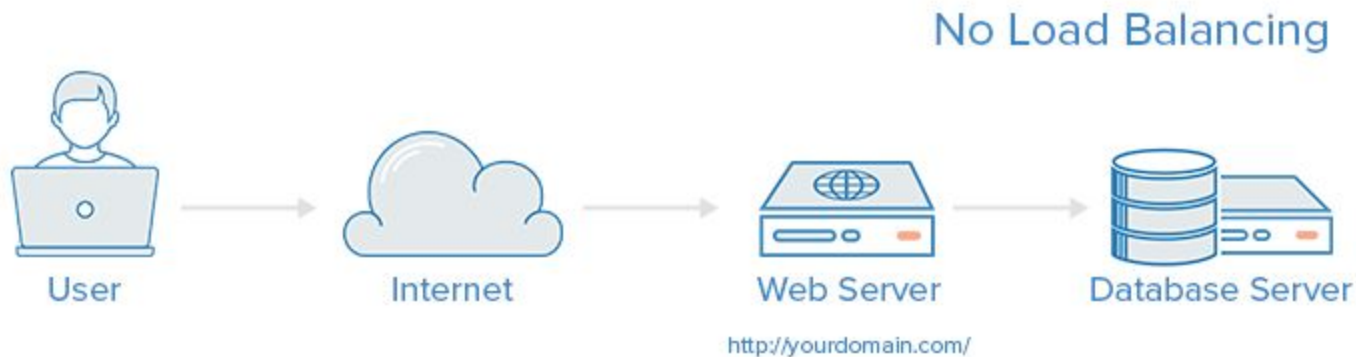
Load Balancer

Retornando ao problema...

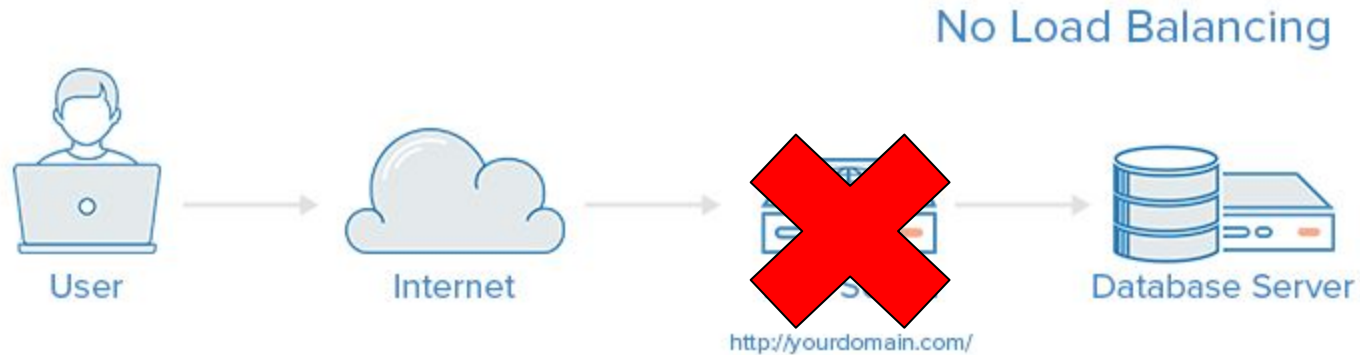
O que aconteceria se na nossa **API de Adesão**, só tivéssemos um servidor atendendo as requisições?



Ou seja... algo parecido com isso



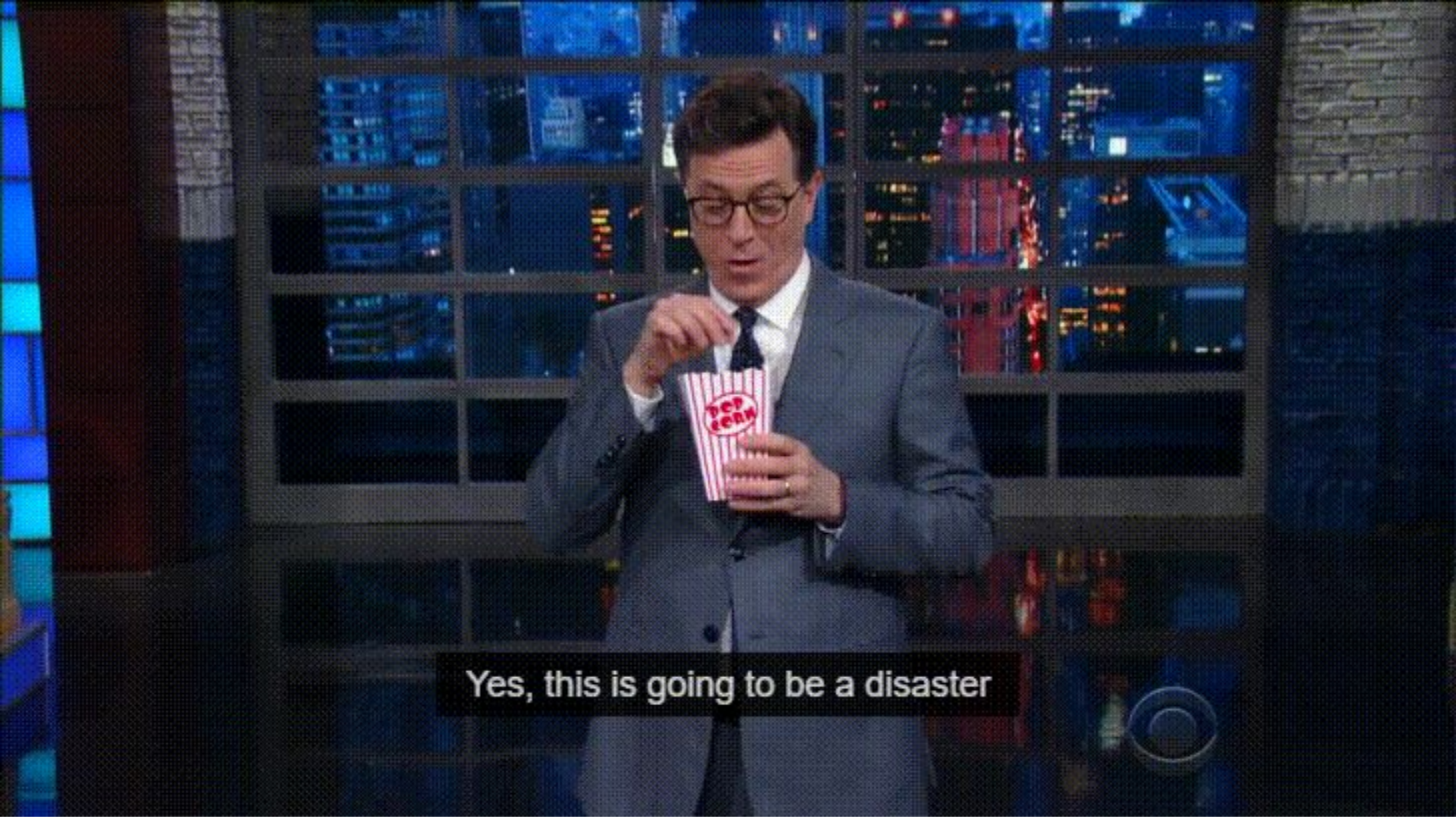
Se essa máquina morrer por qualquer motivo?



Até porque isso...

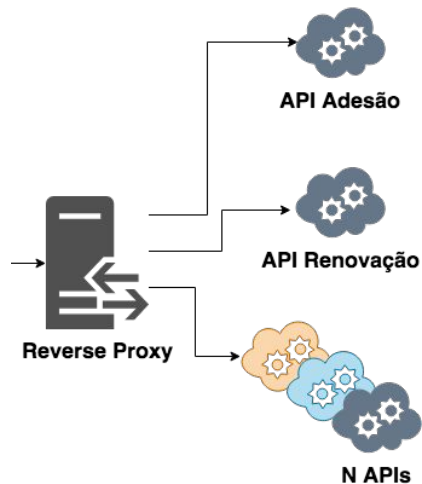


...acontece o tempo todo!

A man with glasses, wearing a grey suit, white shirt, and dark tie, is holding a red and white striped popcorn bucket. He is looking down at the bucket with a concerned expression. The background is a large window showing a city skyline at night with blue and red lights. A CBS eye logo is visible in the bottom right corner.

Yes, this is going to be a disaster

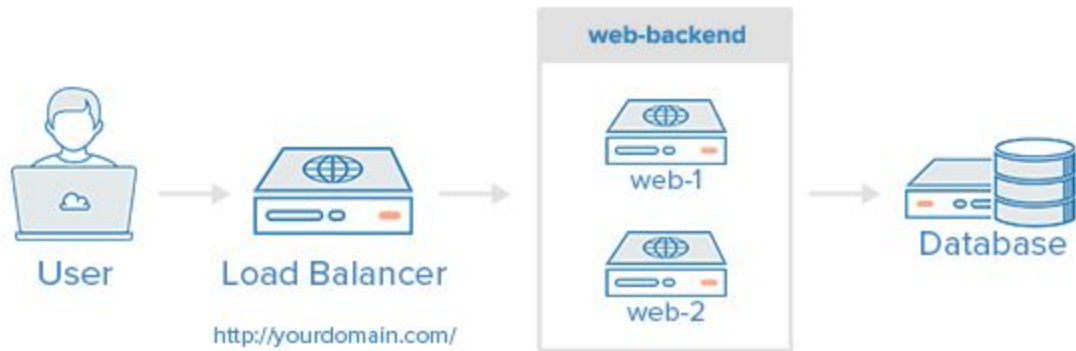
Load Balancer



- Como garantir minimamente minha disponibilidade?
 - Como garantir que se meu servidor morrer minha aplicação não vai cair?
 - Como diminuir as chances de não atender requisições devido momentos de pico de acesso?
 - Ou até mesmo... como controlar a porcentagem de requisições que vão para uma determinada máquina?
-

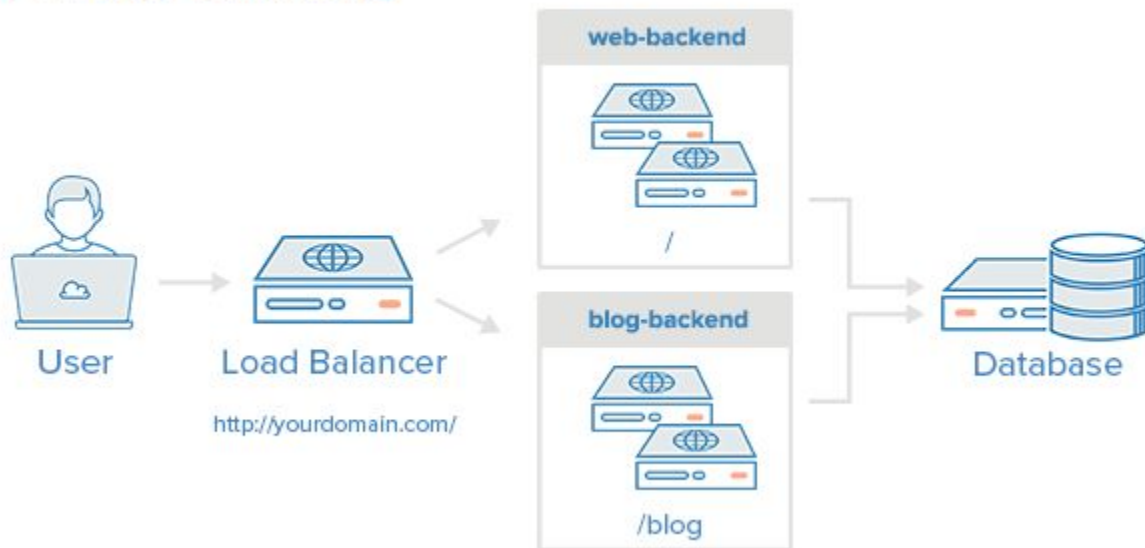
Load Balancer

Layer 4 Load Balancing



Load Balancer

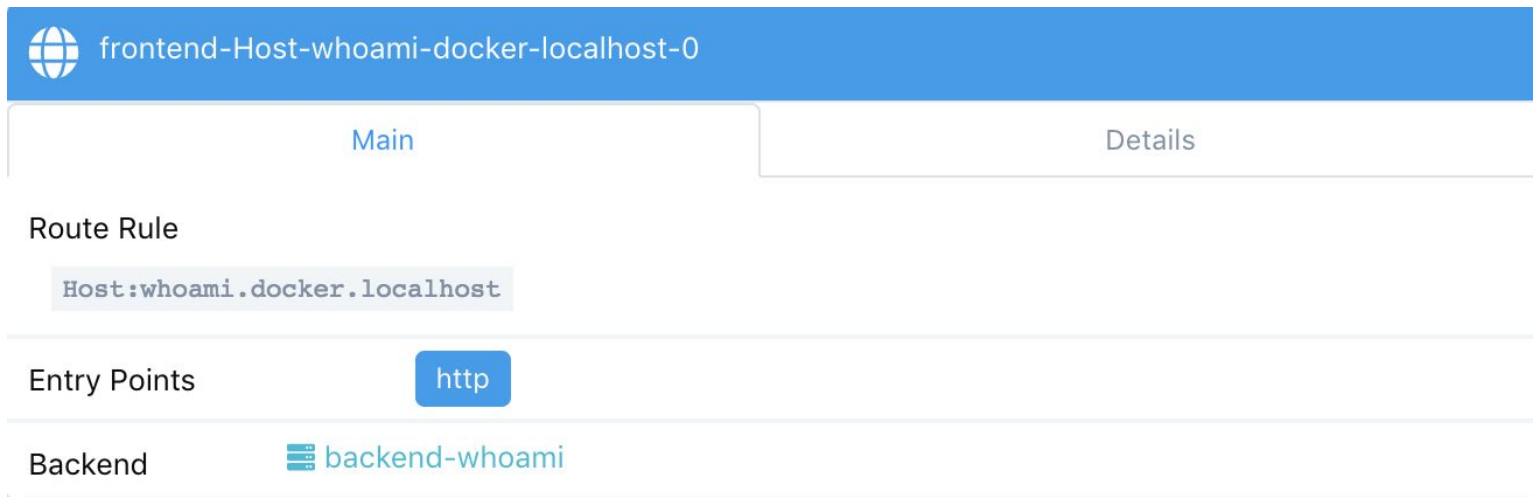
Layer 7 Load Balancing



Load Balancer - Terminologia

Frontends

Conjunto de regras que determinam como as requisições de entrada serão redirecionadas para o **backend**.



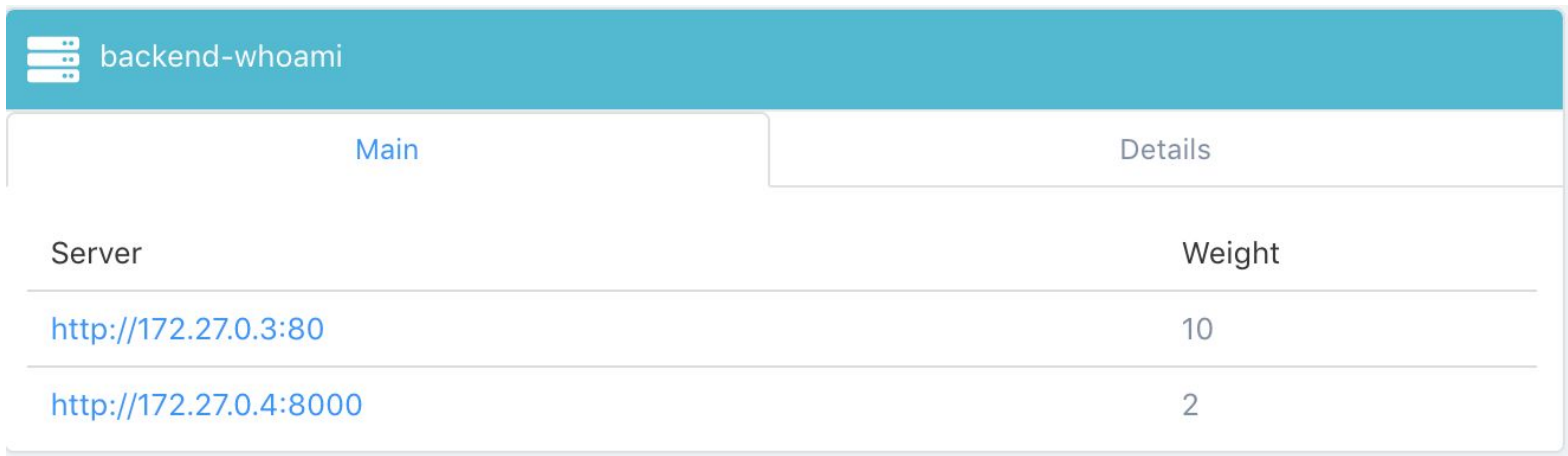
The screenshot shows a configuration page for a frontend named "frontend-Host-whoami-docker-localhost-0". The page has a blue header with a globe icon and the name. Below the header, there are two tabs: "Main" (selected) and "Details". The "Main" tab contains the following configuration:

- Route Rule:** `Host:whoami.docker.localhost`
- Entry Points:** `http`
- Backend:** `backend-whoami`

Load Balancer - Terminologia

Backends

- Conjunto de servidores que recebem os requests redirecionados.
- É o ponto onde haverá de fato o balanceamento da carga



The screenshot shows a web interface for configuring a load balancer. At the top, there is a teal header with a menu icon and the text 'backend-whoami'. Below the header, there are two tabs: 'Main' (selected) and 'Details'. The 'Main' tab displays a table with two columns: 'Server' and 'Weight'. The table contains two rows of data.

Server	Weight
http://172.27.0.3:80	10
http://172.27.0.4:8000	2

—

**Mas vem a velha
pergunta: Por que
mudar?**

Algumas soluções existentes

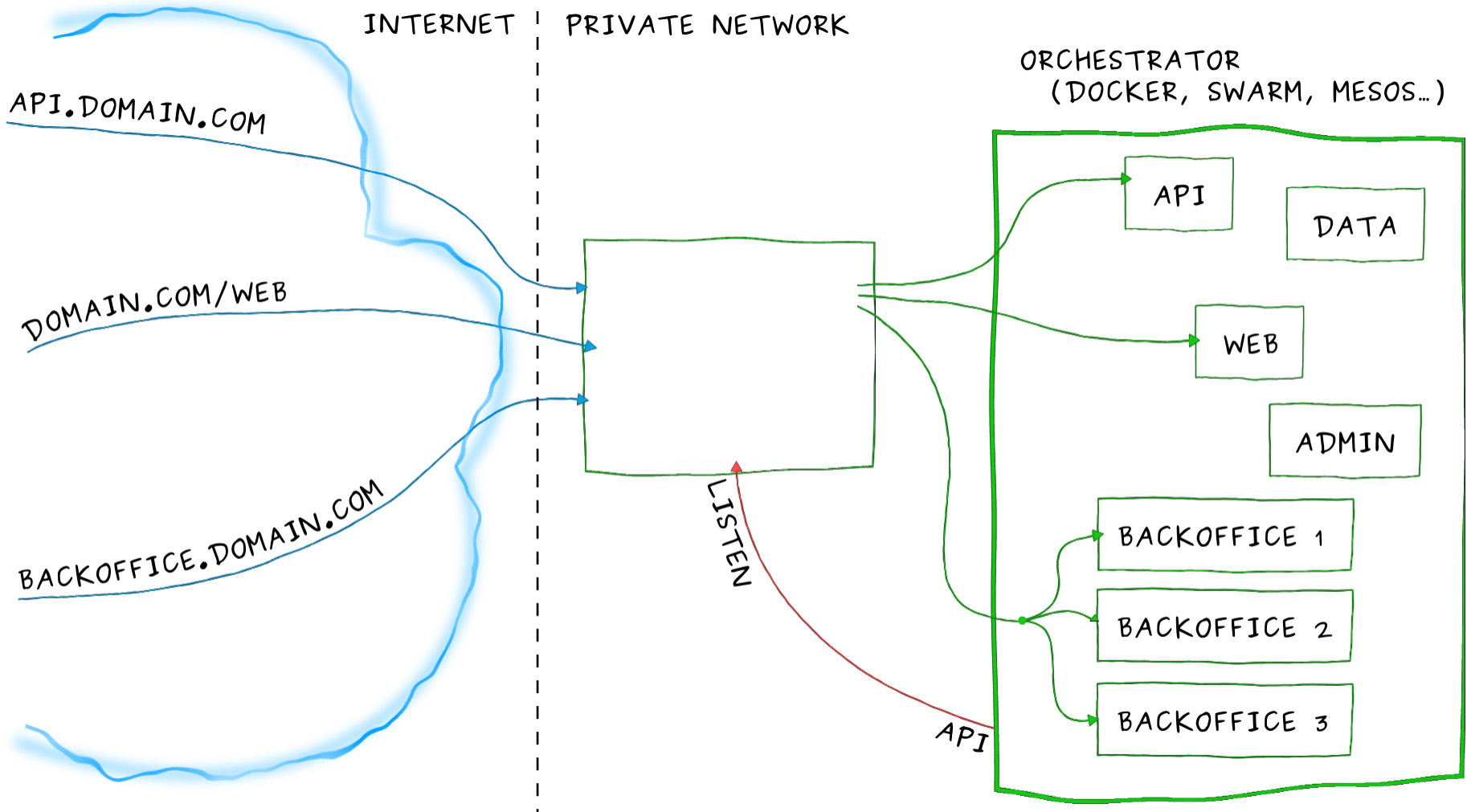


Até atendem a necessidade de reverse proxy e load-balance



Algumas soluções existentes

- Não são dinâmicos - **vários restarts**
 - Configuração complicada - **arquivos conf gigantes**
 - Demandam muita infra - **alto footprint**
 - Difícil manutenção
 - Atendem a uma solução muito específica
 - Pagos
 - etc... etc... etc
-

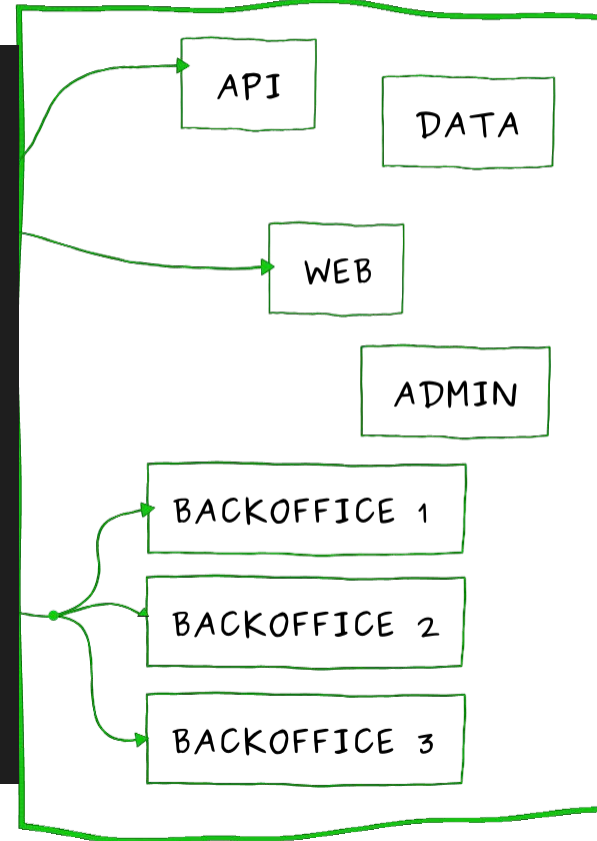


INTERNET | PRIVATE NETWORK

API.DOMAIN.COM

```
4 server:
5   port: 8080
6
7 hystrix:
8   command:
9     default:
10      execution:
11        isolation:
12          thread:
13            timeoutInMilliseconds: 120000
14
15 ribbon:
16   ReadTimeout: 130000
17   ConnectTimeout: 130000
18
19 ribbon.eureka.enabled: false
20
21 blue: #serviceId
22   ribbon:
23     ListOfServers: HOST_IP:PORT_BLUE #list of host:port separated by comma
24     ServerListRefreshInterval: 15000
25
26 green: #serviceId
27   ribbon:
28     ListOfServers: HOST_IP:PORT_GREEN #list of host:port separated by comma
29     ServerListRefreshInterval: 15000
30
31 zuul:
32   ignoredPatterns: /refresh/**
33   routes:
34     main:
35       path: /**
36       strip-prefix: false
37       service-id: CURRENT_COLOR #defaultServiceId
38
39 rps:
40   filter:
41     http-request-filter:
42       enabled: ENABLE_FILTER
43       paths:
44         - regex-path: ^/appV/contexto/.*
45           not-match-service-id: CURRENT_COLOR
46           rules:
47             - match-service-id: NEXT_COLOR
48               methods:
49                 - regex-value: (POST|GET)
50         - regex-path: ^/appV/v1V/contextoV/consulta.*
51           not-match-service-id: CURRENT_COLOR
52           rules:
53             - match-service-id: NEXT_COLOR
54               query-params:
55                 - name: telephone
```

ORCHESTRATOR
(DOCKER, SWARM, MESOS...)





Noooooooooooo

—

**E para resolver
esses problemas
temos...**



træfik



The Cloud Native Edge Router

A reverse proxy / load balancer that's easy, dynamic, automatic, fast, full-featured, open source, production proven, provides metrics, and integrates with every major cluster technology... No wonder it's so popular!

[Download](#)[Get Started](#)

 290M+
Downloads

 18K+
Stars on GitHub

 300+
Contributors

Features That Matter



Auto Discovery

Kubernetes, Mesos,
Docker Swarm, Marathon,
Rancher, ...



Tracing

Open Tracing, Jaeger &
Zipkin



Metrics

Prometheus, Data Dog,
StatsD, InfluxDB, ...



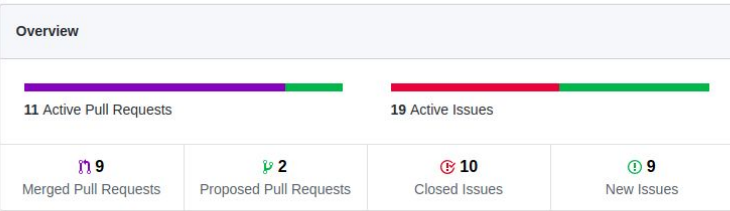
HTTPS

Let's Encrypt, ACME,
custom certificates, ...

- Pulse
- Contributors
- Community
- Commits
- Code frequency
- Dependency graph
- Network
- Forks

November 8, 2018 – November 15, 2018

Period: 1 week

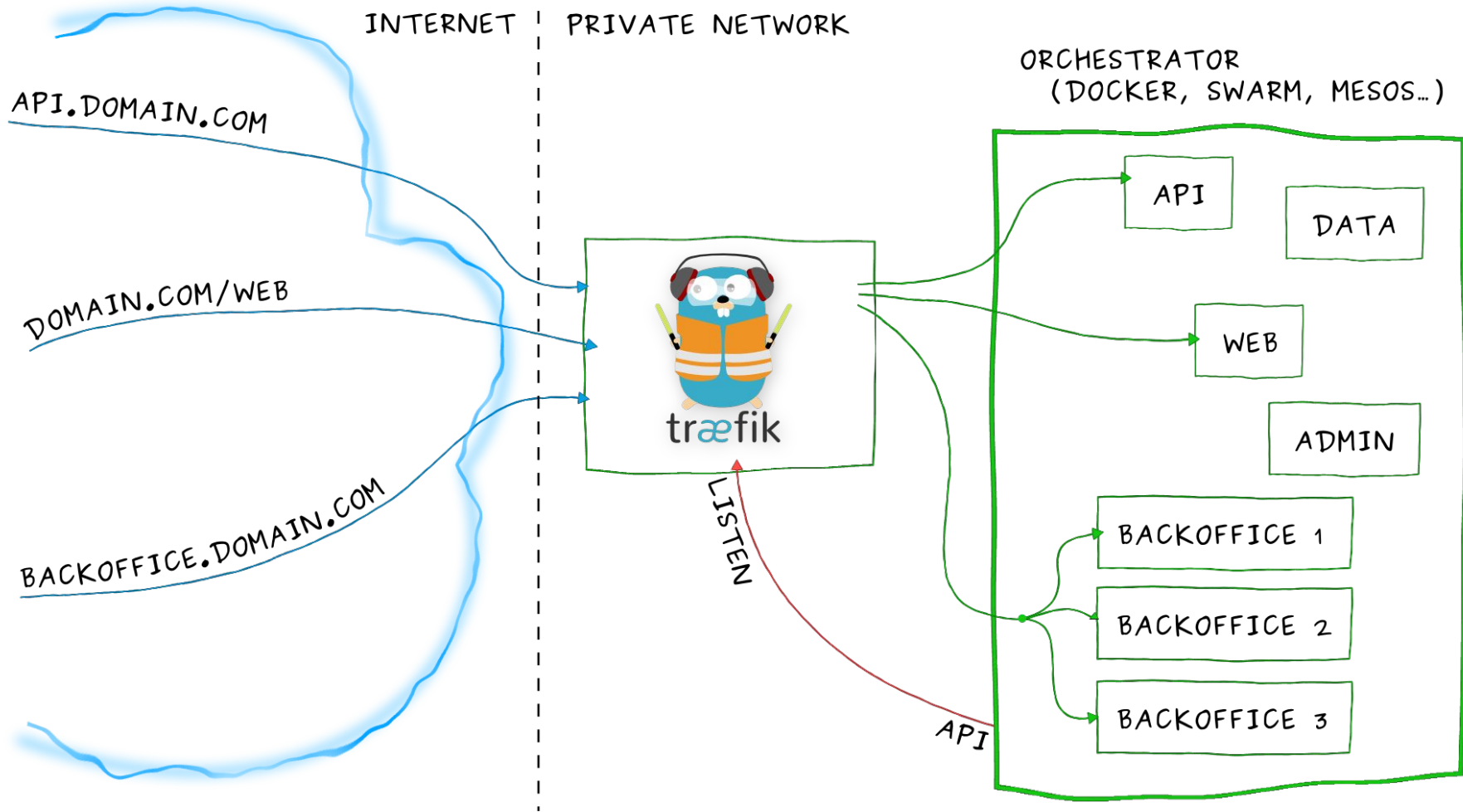


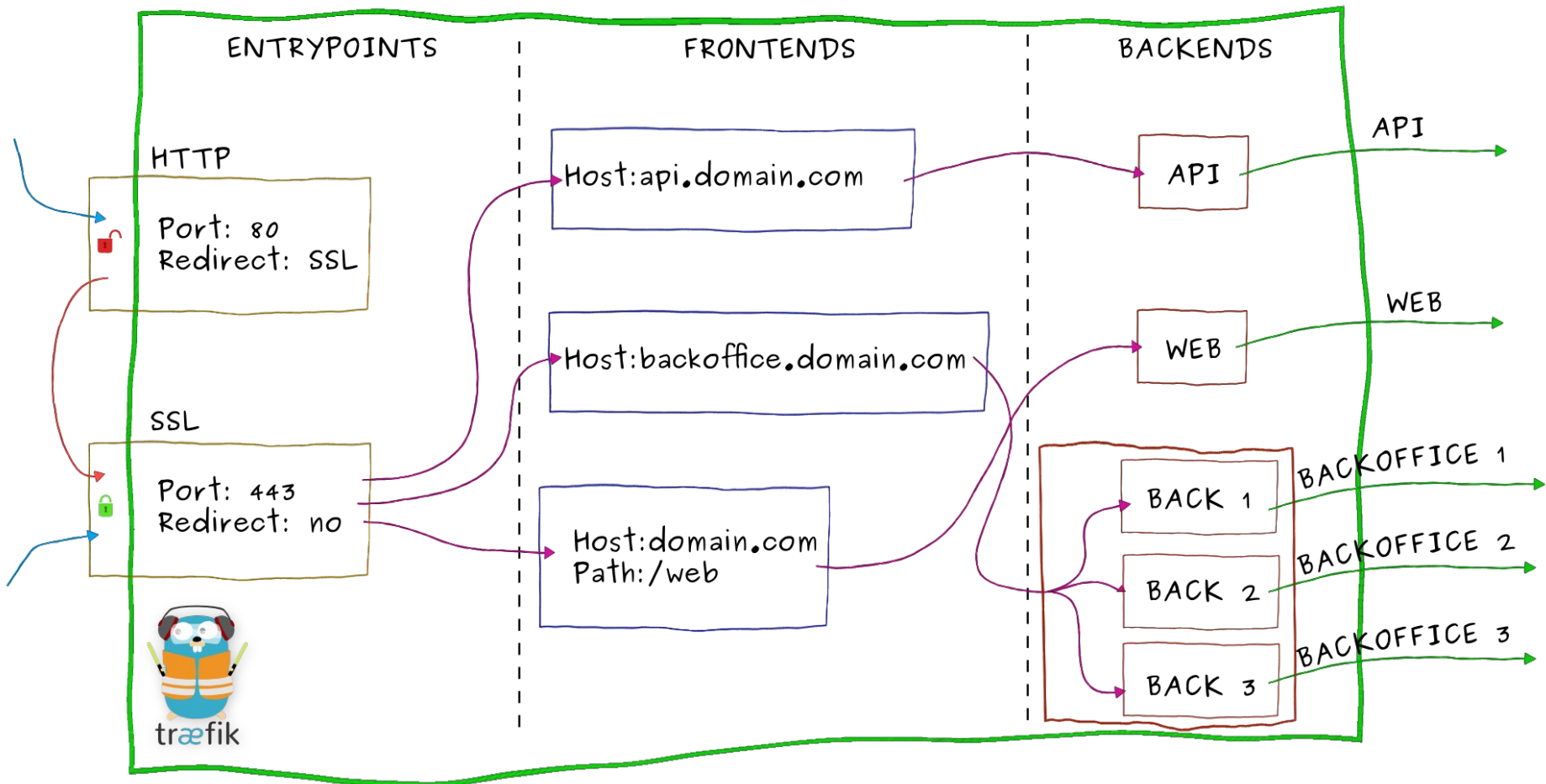
Excluding merges, **10 authors** have pushed **1 commit** to master and **10 commits** to all branches. On master, **452 files** have changed and there have been **20,966 additions** and **9,362 deletions**.



9 Pull requests merged by 9 people

- Merged #4188 Query params in health check 2 hours ago
- Merged #4189 Upgraded DD APM library 6 hours ago
- Merged #4170 Include an explicit list of kubernetes protocol annotations in docs. 6 hours ago
- Merged #4175 Improve kubernetes TLS user guide a day ago
- Merged #4168 Dynamic Configuration Refactoring a day ago
- Merged #4171 frame-deny should be set to true to enable the header a day ago





Funcionalidades

-
- Apenas um pequeno e único binário
 - Hot reloading... sem restarts
 - Load-balancing
 - Circuit Brakers
 - Suporte a Websockets
 - HTTP2
 - Healthchecks
 - Monitoramento

Instalação

Via Download

Baixar a release mais recente...

<https://github.com/containous/traefik/releases>

Execute...

```
$ ./traefik --configFile=traefik.toml
```

Gerenciador de pacote

Via Helm (❤️Kubernetes)

```
$ helm install stable/traefik
```

Via Homebrew

```
$ brew install traefik
```

Docker e Docker Compose

```
$ docker run -d -p 8080:8080 -p 80:80 -v $PWD/traefik.toml:/etc/traefik/traefik.toml traefik
```

ou...

```
version: '3'

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


\$ docker-compose up -d



🔍 Filter by name or id ...

[docker](#)

1 FRONTENDS


 frontend-Host-reverse-proxy-workspace-traefik-presentation-0

Main Details

Route Rule

```
Host:reverse-proxy.workspace-traefik-presentation
```

Entry Points [http](#)

Backend  backend-reverse-proxy-workspace-traefik-presentation

1 BACKENDS

 backend-reverse-proxy-workspace-traefik-presentation

Main Details


Server	Weight
http://172.22.0.2:80	1

Recursos de máquina

```
→ workspace-traefik-presentation docker ps
CONTAINER ID   IMAGE     COMMAND                  CREATED        STATUS        PORTS                                                                 NAMES
0819861de701   traefik   "/traefik --api --do..." 10 minutes ago Up 6 minutes   0.0.0.0:80->80/tcp, 0.0.0.0:8080->8080/tcp traefik

→ workspace-traefik-presentation docker images
REPOSITORY    TAG       IMAGE ID       CREATED        SIZE
traefik        latest   d048a102127b  2 weeks ago   68.8MB
```

68.8MB de espaço em disco

NAME	CID	CPU	MEM
 traefik	0819861de701		0% 17M / 1.95G

Consumo inicial de ~17MB de memória

... da PoC

Prova de conceito

Que comece a mágica...

Vamos adicionar este serviço ao arquivo docker-compose.yml

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

\$ docker-compose up -d whoami

[docker](#)

2 FRONTENDS

frontend-Host-reverse-proxy-workspace-traefik-presentation-1

Main Details

Route Rule

```
Host:reverse-proxy.workspace-traefik-presentation
```

Entry Points [http](#)

Backend backend-reverse-proxy-workspace-traefik-presentation

frontend-Host-whoami-docker-localhost-0

Main Details

Route Rule

```
Host:whoami.docker.localhost
```

Entry Points [http](#)

Backend backend-whoami-workspace-traefik-presentation

2 BACKENDS

backend-reverse-proxy-workspace-traefik-presentation

Main Details

Server	Weight
http://172.22.0.2:80	1

backend-whoami-workspace-traefik-presentation

Main Details

Server	Weight
http://172.22.0.3:80	1



Version: '3'

services:

reverse-proxy:

container_name: traefik

image: traefik # The official Traefik docker image

command: --api --docker # Enables the web UI and tells Traefik to listen to docker

ports:

- "80:80" # The HTTP port
- "8080:8080" # The Web UI (enabled by --api)

volumes:

- /var/run/docker.sock:/var/run/docker.sock # So that Traefik can listen to the Docker events

whoami:

image: containous/whoami # A container that exposes an API to show its IP address

labels:

- "traefik.frontend.rule=Host:whoami.docker.localhost"
- "traefik.backend=whoami"
- "traefik.weight=10"

whoami-green:

image: containous/whoami # A container that exposes an API to show its IP address

labels:

- "traefik.frontend.rule=Host:whoami.docker.localhost"
- "traefik.backend=whoami"
- "traefik.weight=2"



docker

3 FRONTENDS

frontend-Host-reverse-proxy-workspace-traefik-presentation-1

Main Details

Route Rule

Host:reverse-proxy.workspace-traefik-presentation

Entry Points [http](#)

Backend [backend-reverse-proxy-workspace-traefik-presentation](#)

frontend-Host-whoami-docker-localhost-0

Main Details

Route Rule

Host:whoami.docker.localhost

Entry Points [http](#)

Backend [backend-whoami](#)

frontend-Host-whoami-docker-localhost-2

Main Details

Route Rule

Host:whoami.docker.localhost

Entry Points [http](#)

Backend [backend-whoami](#)

2 BACKENDS

backend-reverse-proxy-workspace-traefik-presentation

Main Details

Server	Weight
http://172.24.0.3:80	1

backend-whoami

Main Details

Server	Weight
http://172.24.0.4:80	10
http://172.24.0.2:80	2

```
→ workspace-traefik-presentation curl -H Host:whoami.docker.localhost http://127.0.0.1
```

```
Hostname: ecc8ebb7ec3f  
IP: 127.0.0.1  
IP: 172.24.0.2  
GET / HTTP/1.1  
Host: whoami.docker.localhost  
User-Agent: curl/7.54.0  
Accept: /*/*  
Accept-Encoding: gzip  
X-Forwarded-For: 172.24.0.1  
X-Forwarded-Host: whoami.docker.localhost  
X-Forwarded-Port: 80  
X-Forwarded-Proto: http  
X-Forwarded-Server: 509f41efb5f3  
X-Real-IP: 172.24.0.1
```

\$ curl -H Host:whoami.docker.localhost http://127.0.0.1

→ workspace-traefik-presentation curl -H Host:whoami.docker.localhost http://127.0.0.1

```
Hostname: ecc8ebb7ec3f
IP: 127.0.0.1
IP: 172.24.0.2
GET / HTTP/1.1
Host: whoami.docker.localhost
User-Agent: curl/7.54.0
Accept: */*
Accept-Encoding: gzip
X-Forwarded-For: 172.24.0.1
X-Forwarded-Host: whoami.docker.localhost
X-Forwarded-Port: 80
X-Forwarded-Proto: http
X-Forwarded-Server: 509f41efb5f3
X-Real-IP: 172.24.0.1
```

→ workspace-traefik-presentation curl -H Host:whoami.docker.localhost http://127.0.0.1

```
Hostname: 55ad26c76ced
IP: 127.0.0.1
IP: 172.24.0.4
GET / HTTP/1.1
Host: whoami.docker.localhost
User-Agent: curl/7.54.0
Accept: */*
Accept-Encoding: gzip
X-Forwarded-For: 172.24.0.1
X-Forwarded-Host: whoami.docker.localhost
X-Forwarded-Port: 80
X-Forwarded-Proto: http
X-Forwarded-Server: 509f41efb5f3
X-Real-IP: 172.24.0.1
```

→ workspace-traefik-presentation curl -H Host:whoami.docker.localhost http://127.0.0.1

```
Hostname: 55ad26c76ced
IP: 127.0.0.1
IP: 172.24.0.4
GET / HTTP/1.1
Host: whoami.docker.localhost
User-Agent: curl/7.54.0
Accept: */*
Accept-Encoding: gzip
X-Forwarded-For: 172.24.0.1
X-Forwarded-Host: whoami.docker.localhost
X-Forwarded-Port: 80
X-Forwarded-Proto: http
X-Forwarded-Server: 509f41efb5f3
X-Real-IP: 172.24.0.1
```



Total Response Time

1 second

Total Code Count

1904Uptime Since
2018-11-16 01:01:58 -02:00**30 minutes**

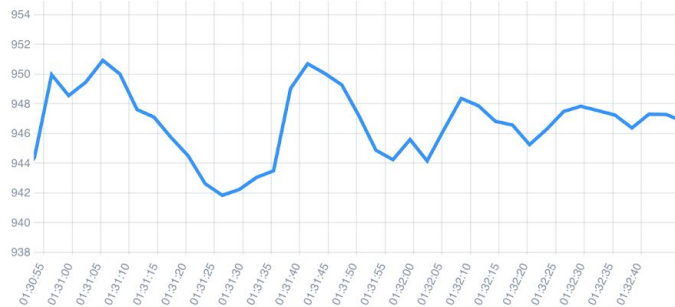
Average Response Time

0 ms

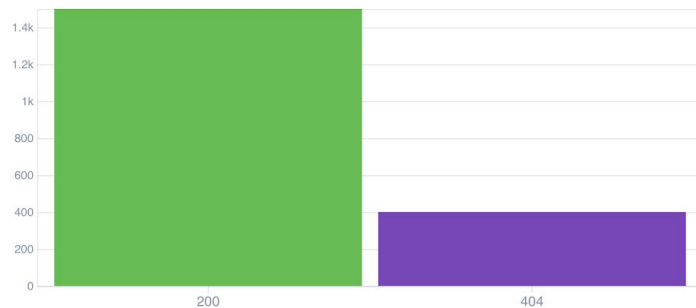
Code Count

0

PID

1Average Response Time (μ s)

Total Status Code Count



Alguns processos em batch estavam rodando

- executando remotamente chamadas de sucesso
- executando remotamente chamadas de falha
- executando local chamadas de sucesso

version: '3'

services:

reverse-proxy:

container_name: traefik

image: traefik # The official Traefik docker image

command: --api --docker # Enables the web UI and tells Traefik to listen to docker

ports:

- "80:80" # The HTTP port
- "8080:8080" # The Web UI (enabled by --api)

volumes:

- /var/run/docker.sock:/var/run/docker.sock # So that Traefik can listen to the Docker events

whoami:

image: containous/whoami # A container that exposes an API to show its IP address

labels:

- "traefik.frontend.rule=Host:whoami.docker.localhost"
- "traefik.backend=whoami"
- "traefik.weight=10"

whoami-green:

image: marcopollivier/whoami

labels:

- "traefik.frontend.rule=Host:whoami.docker.localhost"
- "traefik.backend=whoami"
- "traefik.weight=2"

<> Code

🔔 Issues 3

🔗 Pull requests 3

📊 Insights

Branch: master ▾ whoami / Dockerfile

Find file

Copy path

👤 Idez New build system (#9)

72d3d9c on 9 Aug

3 contributors 

13 lines (11 sloc) | 335 Bytes

Raw

Blame

History



```

1 FROM golang:1.10 as builder
2 WORKDIR /go/src/github.com/containous/whoami
3 COPY . .
4 RUN go get -u github.com/golang/dep/cmd/dep
5 RUN make dependencies
6 RUN make build
7
8 # Create a minimal container to run a Golang static binary
9 FROM scratch
10 COPY --from=builder /go/src/github.com/containous/whoami/w
11 ENTRYPOINT ["/whoami"]
12 EXPOSE 80

```

🔗 marcopollivier / whoami

forked from jwilder/whoami

🔔 Unwatch ▾

1

★ Star

0

Fork

54

<> Code

🔗 Pull requests 0

📁 Projects 0

📖 Wiki

📊 Insights

⚙ Settings

Branch: master ▾ whoami / Dockerfile

Find file

Copy path

👤 ph-One Use Multistage build and Alpine

0a73396 on 10 Jun 2017

2 contributors 

12 lines (10 sloc) | 185 Bytes

Raw

Blame

History



```

1 FROM golang:alpine3.6 AS binary
2 ADD . /app
3 WORKDIR /app
4 RUN go build -o http
5
6 FROM alpine:3.6
7 WORKDIR /app
8 ENV PORT 8000
9 EXPOSE 8000
10 COPY --from=binary /app/http /app
11 CMD ["/app/http"]

```



backend-whoami

Main

Details

Server

Weight

<http://172.27.0.3:80>

10

<http://172.27.0.4:8000>

2

```
→ workspace-traefik-presentation curl -H Host:whoami.docker.localhost http://127.0.0.1
```

```
Hostname: 1106411cc89e  
IP: 127.0.0.1  
IP: 172.25.0.2  
GET / HTTP/1.1  
Host: whoami.docker.localhost  
User-Agent: curl/7.54.0  
Accept: */*  
Accept-Encoding: gzip  
X-Forwarded-For: 172.25.0.1  
X-Forwarded-Host: whoami.docker.localhost  
X-Forwarded-Port: 80  
X-Forwarded-Proto: http  
X-Forwarded-Server: 3fee73d25a3e  
X-Real-Ip: 172.25.0.1
```

```
→ workspace-traefik-presentation curl -H Host:whoami.docker.localhost http://127.0.0.1
```

```
I am 0c181570e7d3 - 2018-11-16 03:43:54.9583662 +0000 UTC m=+268.112691401
```

```
→ workspace-traefik-presentation curl -H Host:whoami.docker.localhost http://127.0.0.1
```

```
Hostname: 1106411cc89e  
IP: 127.0.0.1  
IP: 172.25.0.2  
GET / HTTP/1.1  
Host: whoami.docker.localhost  
User-Agent: curl/7.54.0  
Accept: */*  
Accept-Encoding: gzip  
X-Forwarded-For: 172.25.0.1  
X-Forwarded-Host: whoami.docker.localhost  
X-Forwarded-Port: 80  
X-Forwarded-Proto: http  
X-Forwarded-Server: 3fee73d25a3e  
X-Real-Ip: 172.25.0.1
```



**Até o momento sem nenhum arquivo de
configuração...**

... e sem restart!

**Ainda tem alguma dúvida sobre
ser rápido?**

Então vamos do zero...



[docker](#) [file](#) [kubernetes](#)

3 FRONTENDS

frontend-potato

Main

Details

Route Rule

Host: potato.docker.local

Entry Points

http

https

Backend

backend-potato

frontend-tomato

Main

Details

Misc.

Priority 1

Host Header true

Redirect

Permanent

to https

Basic Authentication

test:\$apr1\$H6uskkkWSlgXLP6ewTrSuBkTrqE8wj/

test2:\$apr1\$d9hr9HBB\$4HxwgUir3HP4EsggP/QNo0

Error Pages

Backend	Query	Status
errorhandler	/[status].html	500 502-504 404
errorhandler2	/[status].html	403 405-407

Whitelist

10.42.0.0/16

152.89.1.33/32

afed:be44::/16

useXForwardedFor true

Headers

Custom Request Headers

3 BACKENDS

backend-potato

Main

Details

Server

Weight

https/172.16.1.2:80

4

https/172.16.1.3:80

2

backend-tomato

Main

Details

Load Balancer

Method wr

Stickiness true

Cookie Name my_cookie

Max Connections

Amount 42

Extractor Function client.ip

Circuit Breaker

Expression NetworkErrorRatio() > 0.5

Health Check

Path /health

Port 80

Interval 10s

Hostname tomato.foobar.com

Buffering

Request Body Bytes

Max 42

Mem 42

Response Body Bytes

Max 42

Mem 42

Retry Expression

IsNetworkError() && Attempts() <= 2

backend-lettuce

Main

Details



Total Response Time

10 hours

Total Code Count

205273

Uptime Since
2018-05-20 20:42:17 +02:00

1 day

Average Response Time

14 ms

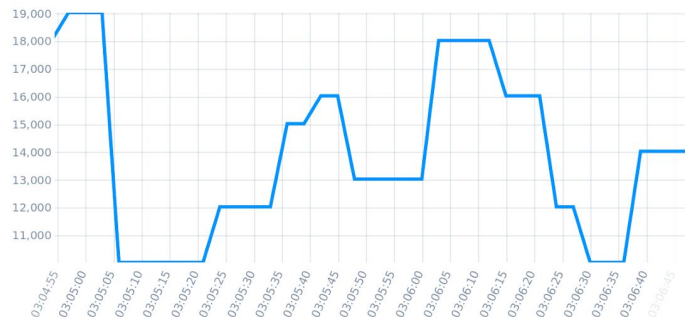
Code Count

0

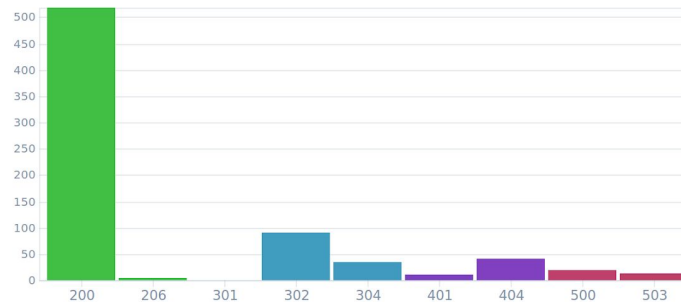
PID

1

Average Response Time (µs)



Total Status Code Count



Recent HTTP Errors

Status	Request	Time
500 Internal Server Error	GET localhost/path	2016-10-22 01:59:15 AM GMT+2
500 Internal Server Error	GET localhost/path	2016-10-22 01:59:15 AM GMT+2
500 Internal Server Error	GET localhost/path	2016-10-22 01:59:15 AM GMT+2
500 Internal Server Error	GET localhost/path	2016-10-22 01:59:15 AM GMT+2

... a Prod

**Ambiente produtivo
e casos de sucesso**

Bacana... funcionou bem em um ambiente controlado, mas a minha infra é mais complexa que isso...

Usando com Swarm mode

```
docker-machine create -d virtualbox mh-keystore
```

```
eval "$(docker-machine env mh-keystore)"  
docker run -d \  
  -p "8500:8500" \  
  -h "consul" \  
  progrium/consul -server -bootstrap
```

Criando uma máquina e instalando o Consul para registrar os serviços

Usando com Swarm mode

```
docker-machine create -d virtualbox \  
  --swarm --swarm-master \  
  --swarm-discovery="consul://$(docker-machine ip mh-keystore):8500" \  
  --engine-opt="cluster-store=consul://$(docker-machine ip mh-keystore):8500" \  
  --engine-opt="cluster-advertise=eth1:2376" \  
mhs-demo0
```



Criando a máquina Swarm master

Usando com Swarm mode

```
docker-machine create -d virtualbox \  
  --swarm \  
  --swarm-discovery="consul://$(docker-machine ip mh-keystore):8500" \  
  --engine-opt="cluster-store=consul://$(docker-machine ip mh-keystore):8500" \  
  --engine-opt="cluster-advertise=eth1:2376" \  
  mhs-demo1
```



Criando a máquina apenas com o agente do Swarm

Usando com Swarm mode

```
eval $(docker-machine env --swarm mhs-demo0)
docker network create --driver overlay --subnet=10.0.9.0/24 my-net
```

Configurações de rede



Usando com Swarm mode

```
→ workspace-traefik-presentation eval $(docker-machine env --swarm mhs-demo0)

→ workspace-traefik-presentation docker run -d --name=whoami0 --net=my-net --env="constraint:node==mhs-demo0" containous/whoami
ec06896f91166e7943e073c3b45241763a8a4abeb9f9b8676222888c38ed2082

→ workspace-traefik-presentation docker run -d --name=whoami1 --net=my-net --env="constraint:node==mhs-demo1" containous/whoami
eaf5ff99368d5b602ecd4df5da465402a694c6c81cbe56ad0cc514329da74dd6
```

Enfim... instalando suas aplicações



🔍 Filter by name or id ...

[docker](#)

4 FRONTENDS

frontend-Host-amazing-johnson-traefik-1

Main Details

Route Rule

`Host:amazing-johnson.traefik`

Entry Points [http](#)

Backend backend-amazing-johnson

frontend-Host-swarm-agent-master-traefik-3

Main Details

Route Rule

`Host:swarm-agent-master.traefik`

Entry Points [http](#)

Backend backend-swarm-agent-master

frontend-Host-swarm-agent-traefik-2

Main Details

Route Rule

`Host:swarm-agent.traefik`

Entry Points [http](#)

Backend backend-swarm-agent

4 BACKENDS

backend-amazing-johnson

Main Details

Server	Weight
http://10.0.9.2:80	1

backend-swarm-agent

Main Details

Server	Weight
http://172.17.0.3:2375	1

backend-swarm-agent-master

Main Details

Server	Weight
http://172.17.0.2:2375	1

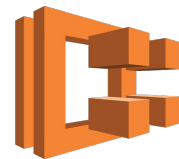
backend-whoami0

Main Details

Server	Weight
http://10.0.9.3:80	1



Algumas soluções suportadas...



AWS ECS



docker



træfik



HashiCorp

Consul



kubernetes



MESOS



Casos de Sucesso

They Trust Traefik for Their Infrastructure

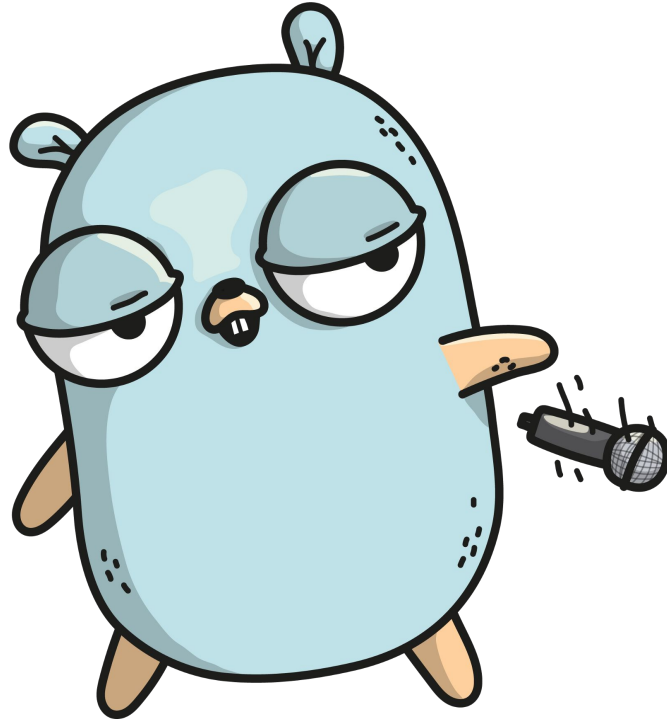



—

Resumindo...

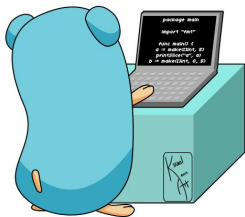
-
- Simples de **instalar**
 - Simples de **configurar**
 - Não precisa **reinstalar**
 - Atende uma ampla gama de soluções de **clusterização e orquestração**
 - Monitoramento **simples**
 - **Leve e rápido**
-

That's It =)



A cartoon illustration of Homer Simpson. He is yellow-skinned with a large nose and wide eyes, wearing a white collared shirt. He is sitting in a brown armchair. Behind him is a pink wall and a window. The window shows a blue sky with white clouds, a blue body of water, and a brown sailboat with two white sails. At the bottom of the image, there is a white text box with a black border containing the text "FIRST OF ALL, THANK YOU." in a bold, sans-serif font.

FIRST OF ALL, THANK YOU.



Comunidade Go



Cassandra Salisbury

@Cassandraoid

⚡ Open Source Strategy @Google ❤️
#DevRel @Golang 🙌 Core Team
@GolangBridge 🐼 #Gopherband

📍 San Francisco, CA

📅 Participa desde setembro de 2014



Daniela (🇧🇷) #elenao

@danicat83

Software developer @ThoughtWorks,
Google Cloud GDE, speaker, blogger,
diversity activist and the proud mother of
two black cats =^.^=

📍 Brasil

🔗 github.com/danicat

📅 Participa desde setembro de 2013



Sheimy

@Sheimy_Rahman Segue você

Software Engineer, C++, Java and
Golang Lover. Oh, Crazy cat Lady, some
times 🐱

📍 127.0.0.1

📅 Participa desde abril de 2018

Família M4U



Daiane Rangel

@daianerangel2 Segue você



Levanto

@caioeverest Segue você

Eu sou muito ruim em escrever bios....
ou qualquer coisa.... me fale de você

📅 Participa desde agosto de 2012



leandro_gomes

@leandro_gomes



Davi Alves

@davidcsalves Segue você



Vini

@http_vini Segue você



Vitor Barbarisi

@VitorBarbarisi Segue você



Felipe Ribeiro

@felipefrbr Segue você



Lilian Feres Ribeiro

@lilianferes! Segue você



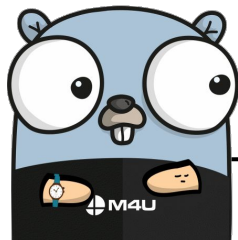
Luiz Picanço

@lpicanco Segue você



Jamal

@soliva! Segue você



Equipe Containous / Træfik



patricia_dugan
@patricia_dugan Segue você

Director of Marketing, @containous.
Biggest fan ever of @traefik. Founder:
@gatherbayarea. 🚴❤️ #pelotonpatricia
#gathersc



Marco Paulo Ollivier @marcopollivier · 2 de nov
Oia nós ae :D

Agradeço muito ao pessoal do @GDGRio por essa oportunidade. Muito feliz.

Palestrante Confirmado! GDG Rio de Janeiro @GDGRio
Palestrante Confirmado!
Marco Ollivier @marcopollivier vem falar sobre Traefik: da PoC a...

1 2 7

patricia_dugan @patricia_dugan **Seguindo**

Em resposta a @marcopollivier @GDGRio

Let us know if you need anything to help support your presentation, Marco! Obrigado!

Traduzir Tweet

22:38 - 3 de nov de 2018 de San Francisco, CA

2 Curtidas

1 2

Saiba mais sobre o Traefik



@traefik



@containous/traefik



traefik.io

—

Perguntas?

Sigam-me :)



@marcopollivier



ollivier.com.br





OH, THAT'S GREAT.
THANK YOU. THANKS A LOT.