

RES

Web Infrastructure Lab

Auteurs :

Simon Baehler

Armand Delessert

Raphaël Racine

Olivier Djeulezeck

Destinataires :

Olivier Liechti

Simon Oulevay

Laurent Prévost

|  |  |
| --- | --- |
| Du | mercredi 6 mai 2015 |
| Au | vendredi 27 mai 2015 |

# Table des matières

[1. Objectifs 2](#_Toc420485598)

[2. Architecture 3](#_Toc420485599)

[3. Développement 4](#_Toc420485600)

[3.1. Reverse proxy 4](#_Toc420485601)

[3.2. Front end 4](#_Toc420485602)

[3.3. Back end 4](#_Toc420485603)

[3.4. Heartbeats du front end et du back end 4](#_Toc420485604)

[3.5. Load balancer 4](#_Toc420485605)

[4. Procédure de validation 4](#_Toc420485606)

# Objectifs

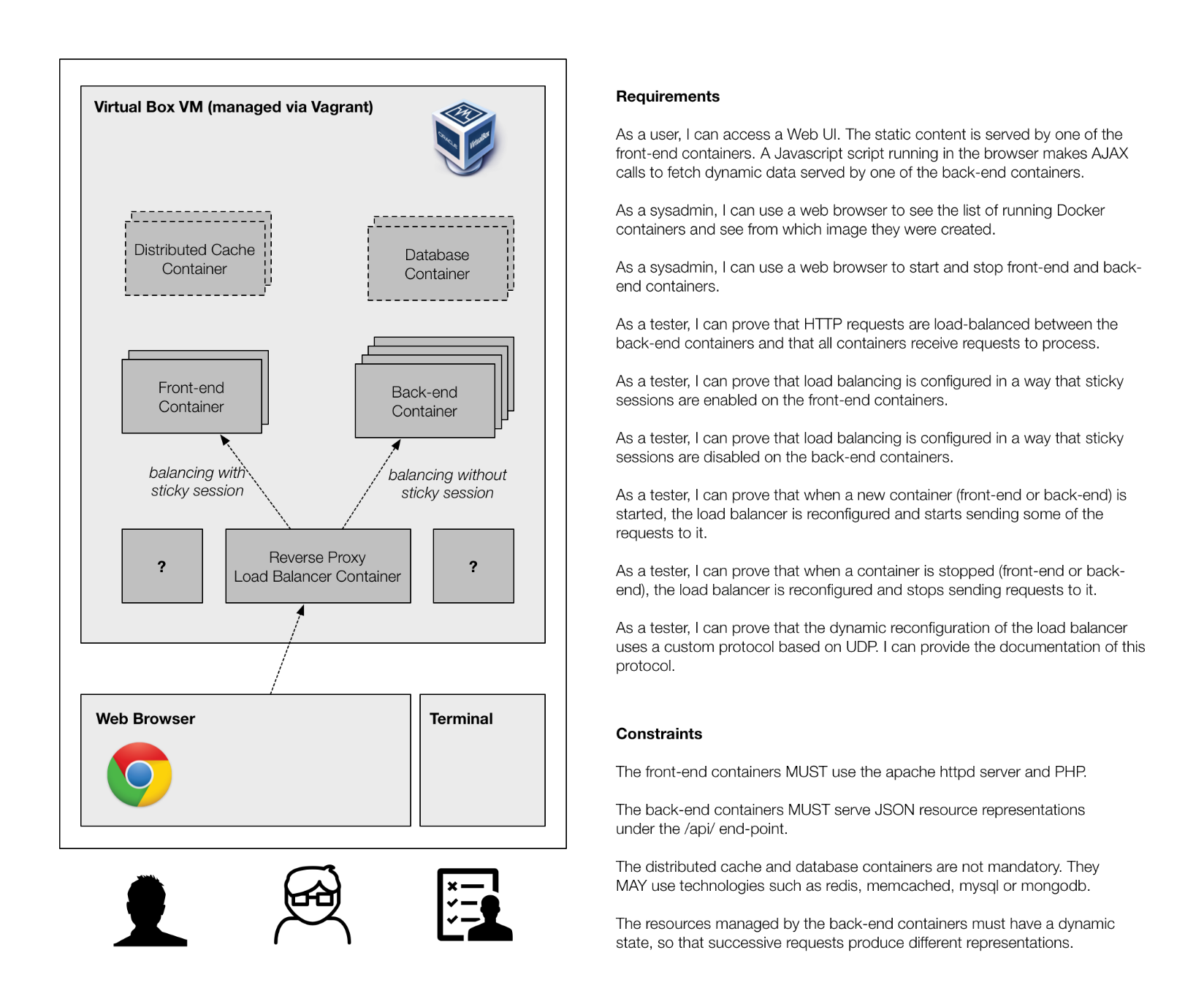
L’objectif principal de ce labo est d’expérimenter la mise en place d’une infrastructure web et de se familiariser avec les différents composants de celle-ci (serveurs http, proxys inverses, équilibreurs de charge).

Le second objectif est de se familiariser avec les outils Vagrant et Docker. En effet, l’infrastructure web à réaliser sera mise en place dans une machine virtuelle Vagrant et via des containers Docker pour chacun des composants de l’infrastructure web.

Enfin, le troisième objectif est d’implémenter la découverte dynamique de service via UDP. Il devra être possible, dans l’infrastrucutre web mise en place, d’ajouter à la volée des serveurs front end ou back end encapsulé dans des containers Docker.

# Architecture

L’architecture de l’infrastructure web à mettre en place est présentée dans l’image suivante.



Le client se connecte à notre service web en entrant l’URL (<http://exemple.com/>) dans son navigateur web. Le navigateur web va alors envoyer une requête GET au reverse proxy qui va la faire suivre au front end et logger l’adresse IP du navigateur web pour les futures connexions de ce dernier. Le front end retourne au navigateur web une page HTML contenant un script JavaScript qui va envoyer des requêtes au back end en s’adressant à l’URL <http://exemple.com/api>.

# Développement

## Back-end

### Structure des fichiers

### Test

## Front-end

### Structure des fichiers

### Test

## Reverse proxy

### Load Balancer

### Sticky Session

## Hearbeat front-end et back-end

# Procédure de validation

Procédure de validation de l’infrastructure. Montrer que ça marche.