DECKHOUSE

Kubernetes Platform

L2 Load Balancer Basics

L2LoadBalancer
powered by MetalLB

front-0   `node-role: front`

                                           nginx-0

front-1   `node-role: front`

                                           nginx-1

front-2   `node-role: front`

                                           nginx-2

worker-0

   app-foo   **app-bar**

There are three frontends nodes and one worker in the cluster.

L2LoadBalancer

powered by MetalLB

```
apiVersion: deckhouse.io/v1alpha1
kind: ModuleConfig
metadata:
  name: l2-load-balancer
spec:
  enabled: true
  settings:
    addressPools:
    - addresses:
      - 192.168.122.0/24
      name: mypool
  version: 1
```
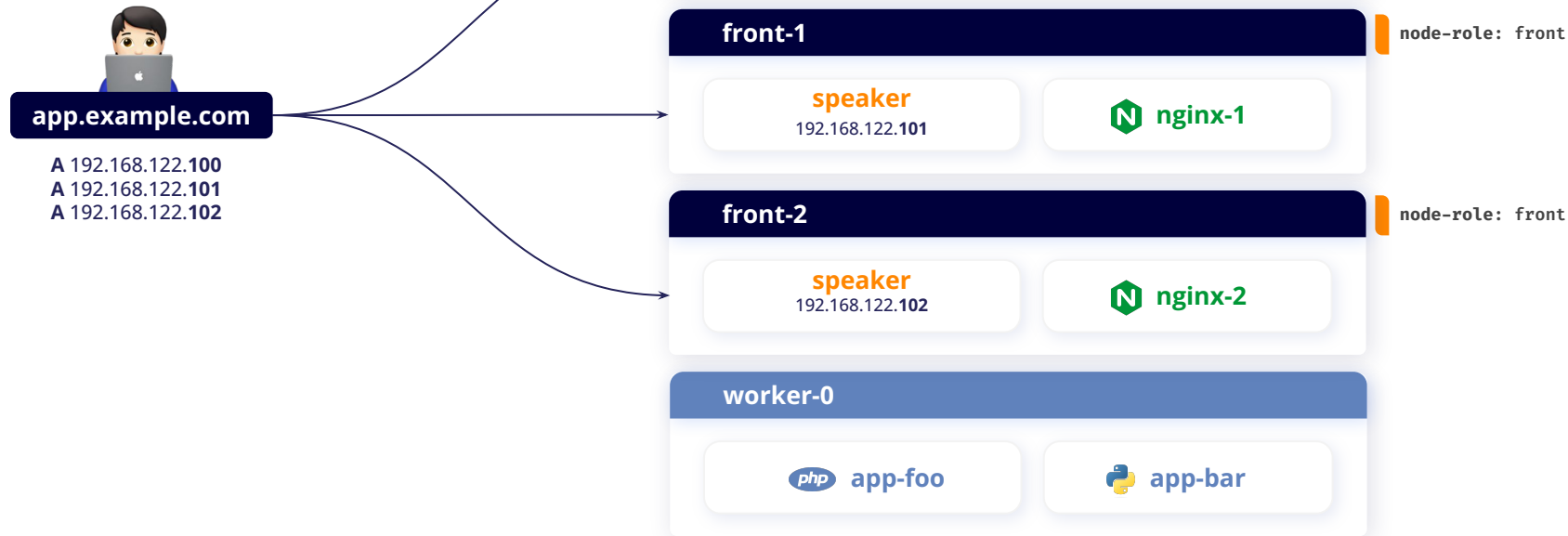
front-0   node-role: front

N nginx-0

front-1   node-role: front

N nginx-1

front-2   node-role: front

N nginx-2

worker-0

php app-foo   app-bar

We turn on L2 Load balancer module with defined address pool.

L2LoadBalancer

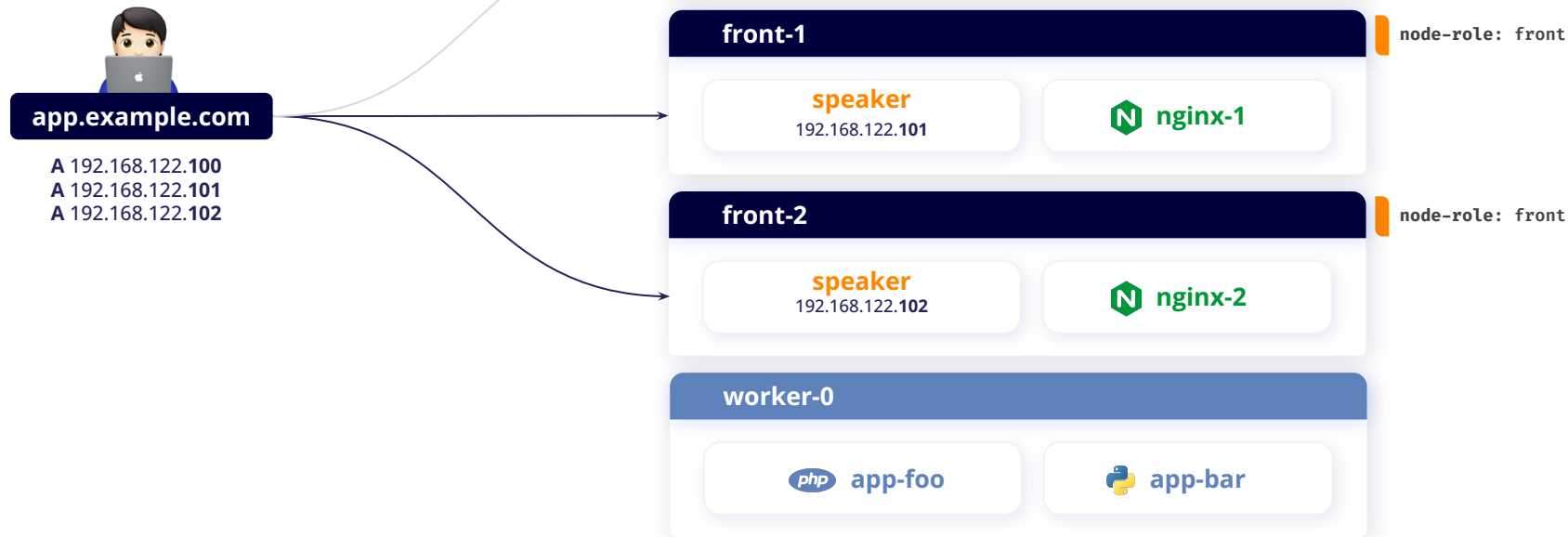powered by MetalLB

```
apiVersion: deckhouse.io/v1alpha1
kind: L2LoadBalancer
metadata:
  name: front
spec:
  addressPool: mypool
  nodeSelector:
    node-role: front
  service:
    selector:
      app: nginx
    ports:
      - name: http
        port: 80
        protocol: TCP
```

front-0

node-role: front

speaker
192.168.122.**100**

nginx-0

front-1

node-role: front

speaker
192.168.122.**101**

nginx-1

front-2

node-role: front

speaker
192.168.122.**102**

nginx-2
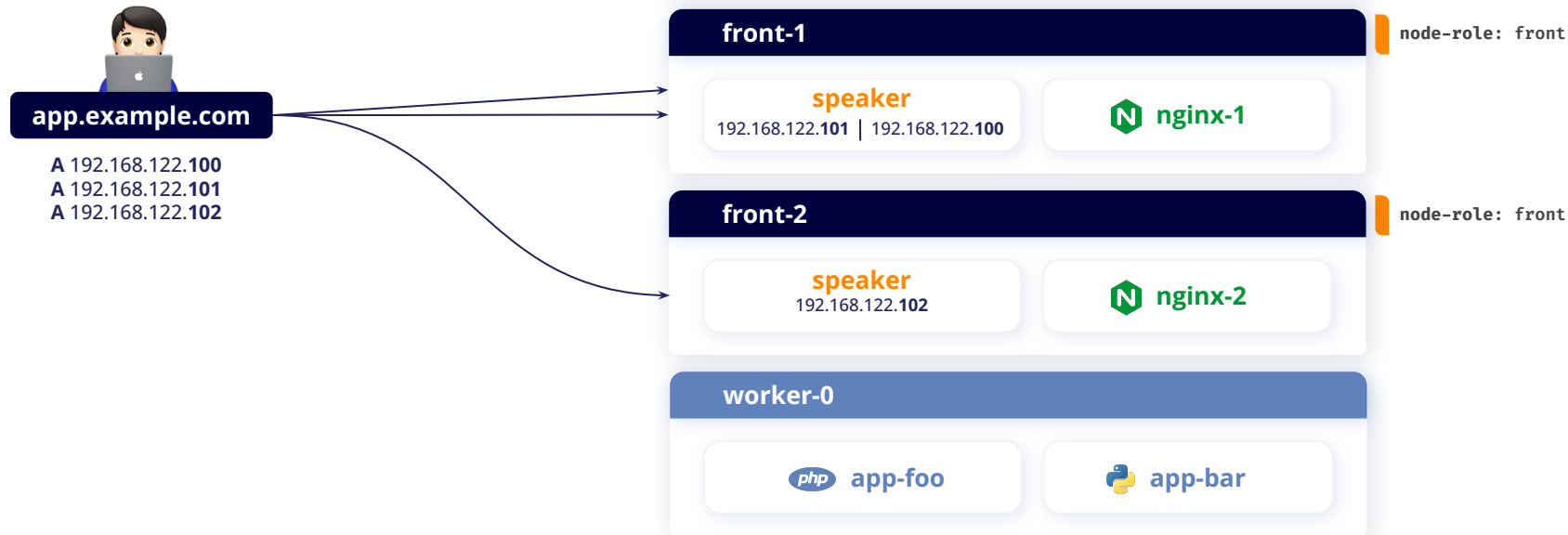
worker-0

php **app-foo**

app-bar

An L2LoadBalancer resource has been created specifying front-end nodes, a pool of "public" IP addresses, and service settings for the nginx application. Speakers are run on all front-end nodes. Each speaker receives a separate address from the pool.

L2LoadBalancer
powered by MetalLB

app.example.com

A 192.168.122.**100**
A 192.168.122.**101**
A 192.168.122.**102**

front-0   node-role: front

speaker
192.168.122.**100**

nginx-0

front-1   node-role: front

speaker
192.168.122.**101**

nginx-1

front-2   node-role: front

speaker
192.168.122.**102**

nginx-2

worker-0

php app-foo

app-bar

Each front-end node participates in handling application requests.
For this, three A records are specified in the public DNS name of the application.

L2LoadBalancer
powered by MetalLB

app.example.com

A 192.168.122.**100**
A 192.168.122.**101**
A 192.168.122.**102**

front-0
node-role: front

speaker
192.168.122.**100**

nginx-0

front-1
node-role: front

speaker
192.168.122.**101**

nginx-1

front-2
node-role: front

speaker
192.168.122.**102**

nginx-2

worker-0

app-foo

app-bar

In the event of a failure of the nginx application on one of the front-end nodes or the node itself, a third of the requests will fail,...

L2LoadBalancer
powered by MetalLB

app.example.com

A 192.168.122.**100**
A 192.168.122.**101**
A 192.168.122.**102**

front-0
node-role: front

speaker

nginx-0

front-1
node-role: front

speaker

192.168.122.**101** | 192.168.122.**100**

nginx-1

front-2
node-role: front

speaker

192.168.122.**102**

nginx-2

worker-0

php app-foo

app-bar

...and one of the remaining front-end nodes will take over the "problematic" IP address and handle the incoming application requests.