

## Balancedores de carga

La siguiente práctica replicará una infraestructura que utiliza un balanceador de carga como servicio (LBaaS) en openstack. La finalidad es proporcionar un servicio para dividir de manera equitativa, peticiones que llegan de diferentes clientes a los servidores virtuales. Específicamente, en esta práctica se crearán 3 servidores que atenderán las peticiones, luego el balanceador distribuirá dichas peticiones. Muy similar al siguiente diagrama:

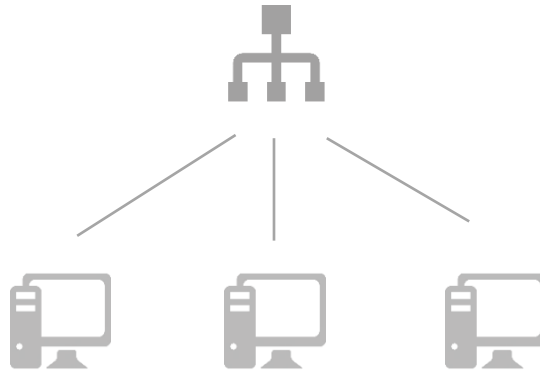


Figura 1: Infraestructura por implementar.

Cada servidor tendrá instalado un servidor Tomcat. Para diferenciar cada servidor, se creará un fichero *index.html* con el número de servidor. De esta manera cada petición que se realice tendrá como respuesta el número de servidor que ha atendido dicha petición. Finalmente, se analizarán las peticiones.

## Procedimiento

- En primer lugar, se creará una red selfservice y una subred dentro de esa red.

```
$ openstack network create selfservice
$ openstack subnet create --subnet-range 10.2.0.0/24 --network selfservice --dns-nameserver 8.8.4.4 subred
```

- Creación de un router que tenga una interfaz (puerta de enlace) a la red externa y otra interfaz a la red creada.

```
$ openstack router create router
$ openstack router add subnet router subred
$ openstack router set router --external-gateway external-network
```

- Se establecen reglas al grupo de seguridad *default* para que se permita el tráfico *HTTP*.

```
$ openstack security group rule create default --protocol tcp --dst-port 80:80 --src-ip 0.0.0.0/0
```

- Seguidamente se crean las instancias que atenderán las peticiones pasando como el archivo `.yaml` a través en el `user-data`.

```
$ NETWORK_ID=$(openstack network list -f value | grep
selfservice | cut -d' ' -f1)
$ openstack server create --image ubuntu-xenial --flavor
lab1 --security-group default --key-name keypair --nic net-
id=$NETWORK_ID --user-data lb1.yaml vmtomcat1

$ openstack server create --image ubuntu-xenial --flavor
lab1 --security-group default --key-name keypair --nic net-
id=$NETWORK_ID --user-data lb2.yaml vmtomcat2

$ openstack server create --image ubuntu-xenial --flavor
lab1 --security-group default --key-name keypair --nic net-
id=$NETWORK_ID --user-data lb2.yaml vmtomcat2
```

- Archivo `yaml`. El texto del archivo `index.html` que crea, se modifica de acuerdo al nodo en el que se implementará.

```
#cloud-config
repo_update: true
packages:
- tomcat8
runcmd:
- [ sh, -c, 'mkdir /var/lib/tomcat8/webapps/id' ]
- [ sh, -c, 'echo "<html><body>Respuesta desde el nodo
2</body></html>" > /var/lib/tomcat8/webapps/id/index.html' ]
- [ sh, -c, 'sudo /sbin/iptables -t nat -I PREROUTING -p tcp -
-dport 80 -j REDIRECT --to-port 8080' ]
```

- Creación de balanceador de carga llamado `web-lb` conectada a la subred `subred`

```
$ neutron lbaas-loadbalancer-create --name web-lb subred
```

- Creación de `listener` para indicar al balanceador de carga qué tipo de tráfico espera atender, en este caso en concreto, tráfico `HTTP`.

```
$ neutron lbaas-listener-create --name web-listener --
loadbalancer web-lb --protocol HTTP --protocol-port 80
```

- Creación del *pool*, que indica los servidores que estarán ligados al balanceador de carga.

```
$ neutron lbaas-pool-create --name web-pool --lb-algorithm
ROUND_ROBIN --listener web-listener --protocol HTTP
```

- Asociación de instancias al *pool* creado previamente

```
$ neutron lbaas-member-create --subnet subred --address 10.2.0.5
--protocol-port 80 web-pool

$ neutron lbaas-member-create --subnet subred --address 10.2.0.7
--protocol-port 80 web-pool

$ neutron lbaas-member-create --subnet subred --address 10.2.0.9
--protocol-port 80 web-pool
```

- Creación de *monitor*, para poder monitorizar periódicamente el estado de cada miembro del *pool*. En los atributos se especifica un *delay*, es decir, el tiempo mínimo en segundos entre las conexiones del miembro. *Type* define el tipo de monitor, específicamente HTTP. *Max-retries* establece el número permitido de conexiones fallidas antes de que el miembro sea considerado inactivo. *Timeout* define el número máximo en segundos para que el monitor espere a que se establezca la conexión antes de que se agote el tiempo, debe ser menor al valor del *delay*.

```
$ neutron lbaas-healthmonitor-create --delay 5 --type HTTP --
max-retries 3 --timeout 2 --pool web-pool
```

- Creación de IP flotante

```
$ openstack floating ip create external-network
```

- Asociación de IP al balanceador de carga.

```
$ FLOATING_IP=$(openstack floating ip list -f value -c "ID")

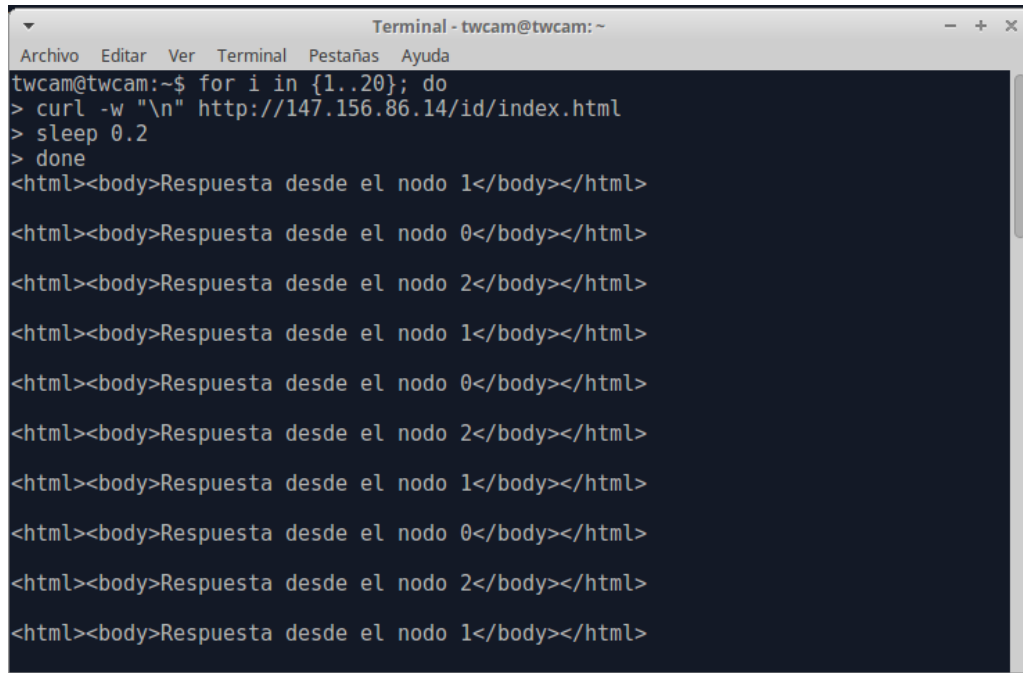
$ PORT_NUM=$(openstack port list | grep loadbalancer | cut -d'
' -f2)
$ neutron floatingip-associate $FLOATING_IP $PORT_NUM
```

- Finalmente, para deshabilitar servidores se ejecuta la siguiente instrucción.

```
$ neutron lbaas-member-update <id-miembro> web-pool --
admin_state_up False
```

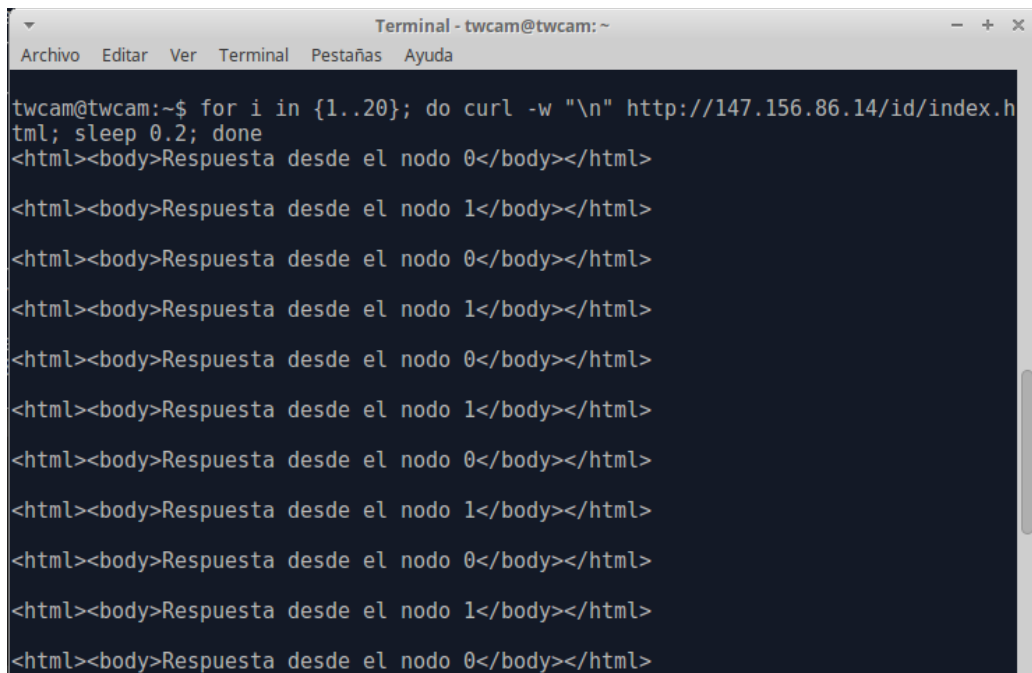
## Pruebas

Al realizar peticiones a la dirección flotante asociada al balanceador de carga, se puede ver que son atendidas por los 3 miembros activos del balanceador.



```
Terminal - twcam@twcam: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
twcam@twcam:~$ for i in {1..20}; do
> curl -w "\n" http://147.156.86.14/id/index.html
> sleep 0.2
> done
<html><body>Respuesta desde el nodo 1</body></html>
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 2</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 2</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 2</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
```

Al deshabilitar un miembro, las peticiones son atendidas únicamente por los miembros activos.



```
Terminal - twcam@twcam: ~
Archivo  Editar  Ver  Terminal  Pestañas  Ayuda
twcam@twcam:~$ for i in {1..20}; do curl -w "\n" http://147.156.86.14/id/index.html; sleep 0.2; done
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
<html><body>Respuesta desde el nodo 0</body></html>
<html><body>Respuesta desde el nodo 1</body></html>
```

Se realizaron 3 pruebas diferentes con 10 usuarios concurrentes haciendo 1000 peticiones.

### 3 servidores activos

Tiempo medio de respuesta	25ms
Desviación	86ms
Rendimiento	307.5/s

Reporte resumen

Nombre:

Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo

Navegar...

Log/Mostrar sólo:

Escribir en Log  Sólo Errores  Éxitos

Configurar

Etiqueta	# Muestras	Media	Min	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
Petición HTTP	10000	25	5	3016	86,27	0,00%	307,5/sec	67,19	65,13	223,8
Total	10000	25	5	3016	86,27	0,00%	307,5/sec	67,19	65,13	223,8

### 2 servidores activos

Tiempo medio de respuesta	28ms
Desviación	123ms
Rendimiento	373.6/s

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo

Navegar...

Log/Mostrar sólo: 

Escribir en Log 

Sólo Errores

Éxitos

Configurar

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
Petición HTTP	10000	28	5	3290	123,35	0,00%	273,6/sec	52,64	57,95	197,0
Total	10000	28	5	3290	123,35	0,00%	273,6/sec	52,64	57,95	197,0

### 1 servidor activo

Tiempo medio de respuesta	27ms
Desviación	83ms
Rendimiento	315.6/s

Reporte resumen

Nombre: Reporte resumen

Comentarios

Escribir todos los datos a Archivo

Nombre de archivo

Navegar...

Log/Mostrar sólo: 

Escribir en Log 

Sólo Errores

Éxitos

Configurar

Etiqueta	# Muestras	Media	Mín	Máx	Desv. Estándar	% Error	Rendimiento	Kb/sec	Sent KB/sec	Media de Bytes
Petición HTTP	10000	28	5	3290	123,35	0,00%	273,6/sec	52,64	57,95	197,0
Total	10000	28	5	3290	123,35	0,00%	273,6/sec	52,64	57,95	197,0