

---

# **Servidores Web de Altas Prestaciones.**

## **Práctica 3**

Balanceo de carga en un sitio web.

Ricardo Ruiz Fernández de Alba

25/05/2023



## Índice

<b>Introducción</b>	<b>2</b>
<b>Tareas</b>	<b>2</b>
<b>Tarea 1. Balanceador de carga</b>	<b>2</b>
Instalación de nginx . . . . .	3
<b>Tarea 2. Alta carga con Apache Benchmark</b>	<b>6</b>
<b>Tarea 3. Análisis Comparativo</b>	<b>6</b>
<b>Referencias</b>	<b>6</b>

## Introducción

En esta práctica, el objetivo es configurar las máquinas virtuales de forma que dos hagan de servidores web finales mientras que la tercera haga de balanceador de carga por software.

## Tareas

En esta práctica se llevarán a cabo las **tareas básicas**:

1. Configurar una máquina e instalar nginx y haproxy como balanceadores de carga con el algoritmo round-robin
2. Someter la granja web a una alta carga con la herramienta Apache Benchmark a través de M3, considerando 2 opciones:
  - a) nginx con round-robin
  - b) haproxy con round-robin
3. Realizar un análisis comparativo de los resultados considerando el número de peticiones por unidad de tiempo

Como **opciones avanzadas**: 1. Configurar nginx y haproxy como balanceadores de carga con ponderación, suponiendo que M1 tiene el doble de capacidad que M2. 2. Habilitar el módulo de estadísticas en HAproxy con varias opciones y analizarlo 3. Instalar y configurar otros balanceadores de carga (Gobetween, Zevenet, Pound, etc.) 4. Someter la granja web a una alta carga con la herramienta Apache Benchmark considerando los distintos balanceadores instalados y configurados. 5. Realizar un análisis comparativo de los resultados considerando el número de peticiones por unidad de tiempo

## Tarea 1. Balanceador de carga

Creamos una nueva máquina virtual llamada m3-ricardoruiz con Ubuntu Server 22.04 LTS, a la que añadiremos el usuario ricardoruiz con contraseña Swap12324.

Configuración de perfil [ Help ]

Proporcione el nombre de usuario y la contraseña que utilizará para acceder al sistema. Puede configurar el acceso SSH en la pantalla siguiente, pero aun se necesita una contraseña para sudo.

Su nombre: Ricardo Ruiz Fernández de Alba

El nombre del servidor: m3-ricardoruiz  
The name it uses when it talks to other computers.

Elija un nombre de usuario: ricardoruiz

Elija una contraseña: \*\*\*\*\*

Confirme la contraseña: \*\*\*\*\*

[ Hecho ]

## Instalación de nginx

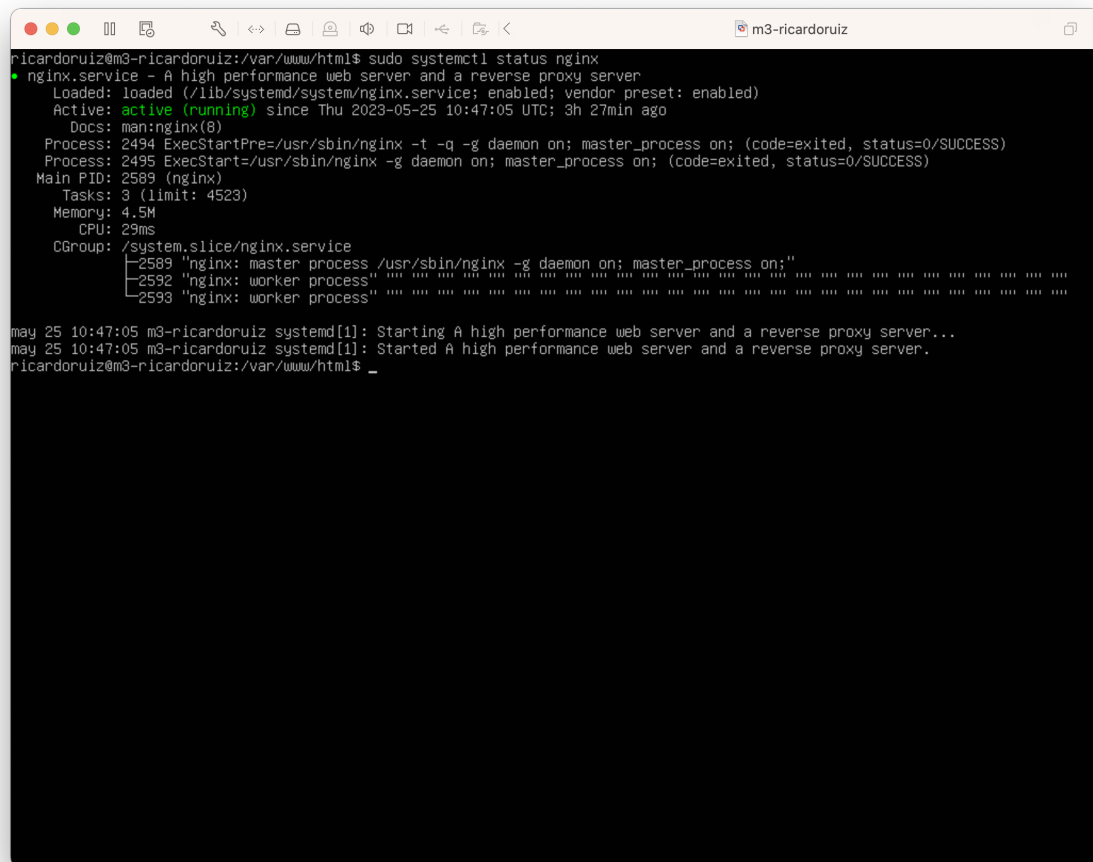
Seguiremos la guía de instalación de nginx para Ubuntu Server 22.04 de Digital Ocean.

```
1 ricardoruiz@m3-ricardoruiz $ sudo apt update
2 ricardoruiz@m3-ricardoruiz $ sudo apt install nginx
```

Antes de probar Nginx, es necesario configurar el firewall para permitir el acceso al servicio. Nginx se registra como un servicio en ufw durante la instalación, lo que facilita permitir el acceso a Nginx.

```
1 ricardoruiz@m3-ricardoruiz $ sudo ufw allow 'Nginx HTTP'
```

Comprobamos que nginx está activo con `sudo systemctl status nginx`:

A terminal window titled 'm3-ricardoruiz' showing the output of the command 'sudo systemctl status nginx'. The output indicates that the nginx service is active (running) and has been loaded. It also shows the process details for the master and worker processes. At the bottom, there are logs from systemd showing the service starting successfully.

```
ricardoruiz@m3-ricardoruiz:/var/www/html$ sudo systemctl status nginx
• nginx.service - A high performance web server and a reverse proxy server
   Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-05-25 10:47:05 UTC; 3h 27min ago
     Docs: man:nginx(8)
   Process: 2494 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Process: 2495 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
  Main PID: 2589 (nginx)
    Tasks: 3 (limit: 4523)
   Memory: 4.5M
      CPU: 29ms
   CGroup: /system.slice/nginx.service
           └─2589 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─2592 "nginx: worker process"
             └─2593 "nginx: worker process"

may 25 10:47:05 m3-ricardoruiz systemd[1]: Starting A high performance web server and a reverse proxy server...
may 25 10:47:05 m3-ricardoruiz systemd[1]: Started A high performance web server and a reverse proxy server.
ricardoruiz@m3-ricardoruiz:/var/www/html$ _
```

**Figura 1:** Nginx

Debemos deshabilitar la configuración por defecto de nginx como servidor web para que actúe como balanceador.

Para ello, comentamos la línea

```
1 #include /etc/nginx/sites-enabled/*;
```

del fichero de configuración `/etc/nginx/nginx.conf`.

Creamos una nueva configuración en `/etc/nginx/conf.d/default.conf`

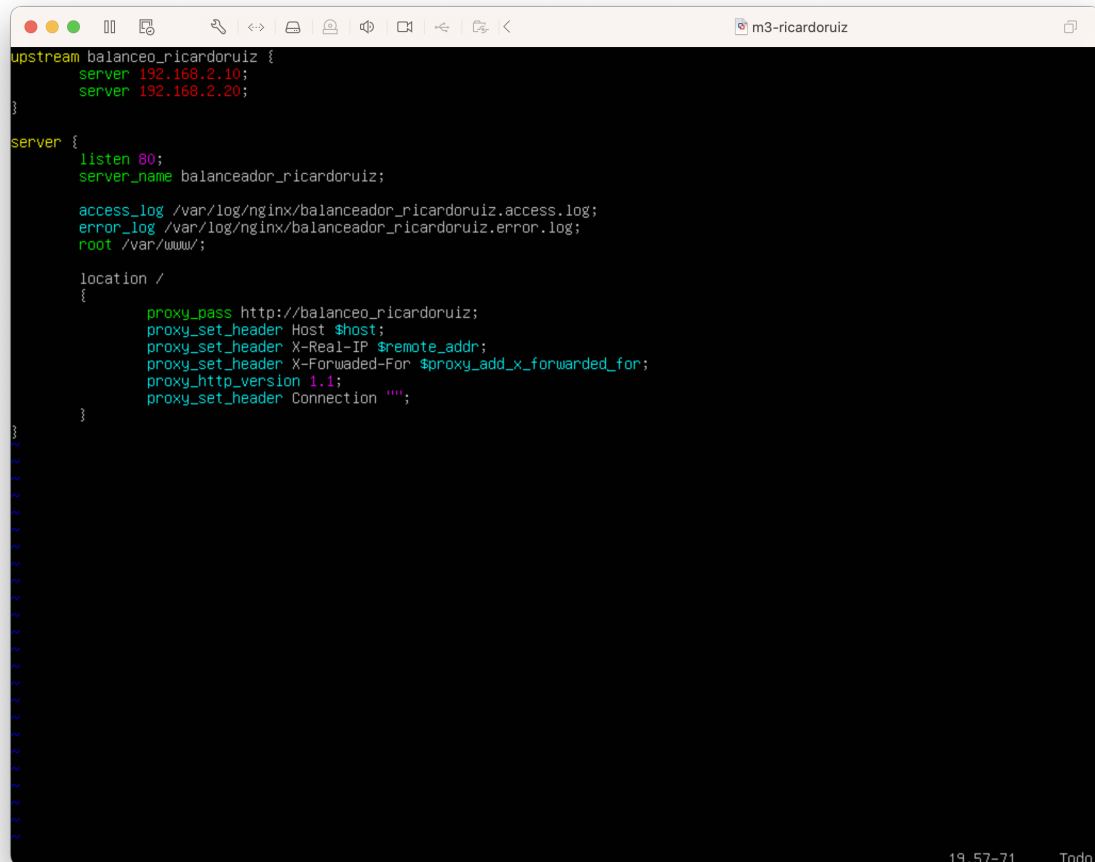
Para definir la granja web de servidores apache escribimos la sección upstream con la IP de las M1 y M2. Es importante que este al principio del archivo de configuración, fuera de la sección server.

```
1 upstream balanceo_ricardoruiz {
2     server 192.168.2.10;
3     server 192.168.2.20;
4 }
```

Debemos definir ahora la sección server para indicar a nginx que use el grupo definido anteriormente en upstream. Para que el proxy\_pass funcione correctamente , debemos indicar una conexión de tipo HTTP 1.1 así como eliminar la cabecera `Connection` para evitar que se pase al servidor final la cabecera que indica el usuario.

```
1 [..]
2 server {
3     listen 80;
4     server_name balanceador_ricardoruiz;
5     access_log /var/log/nginx/balanceador_ricardoruiz.access.log;
6     error_log /var/log/nginx/balanceador_ricardoruiz.error.log;
7     root /var/www/;
8     location / {
9         proxy_pass http://balanceo_ricardoruiz;
10        proxy_set_header Host $host;
11        proxy_set_header X-Real-IP $remote_addr;
12        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
13        proxy_http_version 1.1;
14        proxy_set_header Connection "";
15    }
16 }
```

Luego la configuración completa quedaría como sigue:



```
upstream balanceo_ricardoruiz {
    server 192.168.2.10;
    server 192.168.2.20;
}

server {
    listen 80;
    server_name balanceador_ricardoruiz;

    access_log /var/log/nginx/balanceador_ricardoruiz.access.log;
    error_log /var/log/nginx/balanceador_ricardoruiz.error.log;
    root /var/www/;

    location /
    {
        proxy_pass http://balanceo_ricardoruiz;
        proxy_set_header Host $host;
        proxy_set_header X-Real-IP $remote_addr;
        proxy_set_header X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_http_version 1.1;
        proxy_set_header Connection "";
    }
}
```

Configuramos la IP de la máquina m3-ricardoruiz como IP estática en el fichero `/etc/netplan/00-installer-config.yaml`:

```
yaml network: version: 2 renderer: networkd ethernets: ens160: dhcp4
: true addresses: - 192.168.1.20/24 routes: - to: 0.0.0.0/0 via:
192.168.1.1 metric: 100 nameservers: addresses: [8.8.8.8, 8.8.4.4]
ens256: dhcp4: false addresses: - 192.168.1.30/24a
```

## Tarea 2. Alta carga con Apache Benchmark

## Tarea 3. Análisis Comparativo

## Referencias