<https://raul-profesor.github.io/DEAW/P1.3/>

#### **Desactivar configuraciones previas**

**Eliminar configuraciones anteriores en el proxy inverso:**bash  
Copiar código  
cd /etc/nginx/sites-enabled

unlink archivo\_antiguo\_1

unlink archivo\_antiguo\_2

#### **Servidor Web 1**

1. **Configurar Nginx en el Servidor Web 1:**

**Crear directorio del sitio web:**bash  
Copiar código  
sudo mkdir -p /var/www/webserver1/html

sudo chown -R $USER:$USER /var/www/webserver1/html

**Crear index.html:**bash  
Copiar código  
echo "<h1>Servidor Web 1</h1>" | sudo tee /var/www/webserver1/html/index.html

**Crear archivo de configuración en sites-available:**bash  
Copiar código  
sudo nano /etc/nginx/sites-available/webserver1

Contenido del archivo:  
bash  
Copiar código  
server {

listen 8080;

server\_name webserver1;

location / {

root /var/www/webserver1/html;

index index.html;

add\_header Serv\_Web1\_vuestronombre;

}

}

**Habilitar la configuración:**bash  
Copiar código  
sudo ln -s /etc/nginx/sites-available/webserver1 /etc/nginx/sites-enabled/

sudo systemctl restart nginx

#### **Servidor Web 2**

1. **Clonar y Configurar Nginx en el Servidor Web 2:**
   * **Clonar máquina Debian (por ejemplo, desde VirtualBox).**
   * **Configurar IP fija en dhcp:**
     + Editar /etc/network/interfaces o configurar IP fija a través del servidor DHCP.
   * **Replicar configuraciones:**

Crear directorio:  
bash  
Copiar código  
sudo mkdir -p /var/www/webserver2/html

sudo chown -R $USER:$USER /var/www/webserver2/html

Crear index.html:  
bash  
Copiar código  
echo "<h1>Servidor Web 2</h1>" | sudo tee /var/www/webserver2/html/index.html

Crear archivo de configuración en sites-available:  
bash  
Copiar código  
sudo nano /etc/nginx/sites-available/webserver2

Contenido del archivo:  
bash  
Copiar código  
server {

listen 8080;

server\_name webserver2;

location / {

root /var/www/webserver2/html;

index index.html;

add\_header Serv\_Web2\_vuestronombre;

}

}

Habilitar configuración:  
bash  
Copiar código  
sudo ln -s /etc/nginx/sites-available/webserver2 /etc/nginx/sites-enabled/

sudo systemctl restart nginx

#### **Configurar Proxy Inverso**

1. **Crear archivo de configuración para balanceo de carga:**

En el proxy inverso, crear el archivo de configuración:  
bash  
Copiar código  
sudo nano /etc/nginx/sites-available/balanceo

Contenido del archivo:  
arduino  
Copiar código  
upstream backend\_hosts {

random;

server IP\_SERVIDOR\_WEB\_1:8080;

server IP\_SERVIDOR\_WEB\_2:8080;

}

server {

listen 80;

server\_name balanceo;

location / {

proxy\_pass http://backend\_hosts;

}

}

* + Sustituir IP\_SERVIDOR\_WEB\_1 y IP\_SERVIDOR\_WEB\_2 por las IP de las máquinas configuradas.

Habilitar la configuración:  
bash  
Copiar código  
sudo ln -s /etc/nginx/sites-available/balanceo /etc/nginx/sites-enabled/

sudo systemctl restart nginx

#### **Comprobaciones**

1. **Desde el navegador de la máquina anfitriona:**
   * Accede a: http://balanceo (asegúrate de configurar /etc/hosts si es necesario para resolver balanceo).
   * **Presionar repetidamente F5**: Las respuestas deben alternar entre los dos servidores. Comprueba la cabecera HTTP (Serv\_Web1\_vuestronombre o Serv\_Web2\_vuestronombre) para confirmar.

#### **Prueba de fallos**

1. **Detener un servidor web:**

En el servidor web 1:  
bash  
Copiar código  
sudo systemctl stop nginx

* + Accede de nuevo a http://balanceo para verificar que todas las solicitudes se redirigen al servidor 2.

1. **Reiniciar el servidor y detener el segundo:**

Iniciar servidor web 1:  
bash  
Copiar código  
sudo systemctl start nginx

* + Detener servidor web 2 y comprobar:  
    bash  
    Copiar código  
    sudo systemctl stop nginx