



Balanceador de carga con Nginx.

- Juan Camilo Gonzalez.
- David Steven Aristizabal.
- Manuel Antonio Tarazona.
- Christiam Fernando Vera.

Objetivo general

Implementar balanceador de carga utilizando la herramienta Nginx donde se utilizara dos máquinas o servidores como backend y un servidor donde se implementara el balanceador de carga.

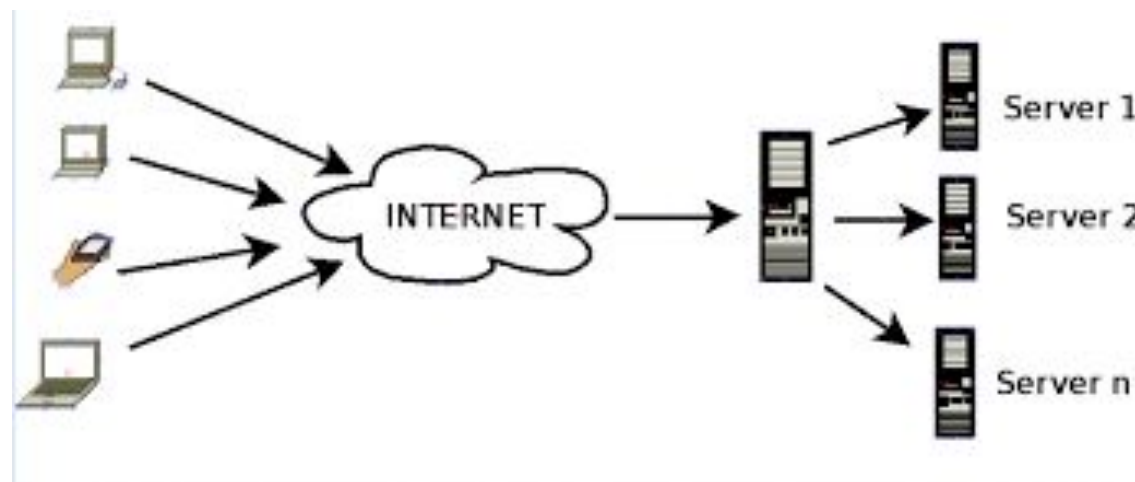
Objetivos específicos

- Configurar Nginx como balanceador de carga.
- Configurar backends con sus respectivas páginas web.
- Configurar DNS para el servidor de frontend.
- Realizar pruebas con Jmeter.

Descripción del problema

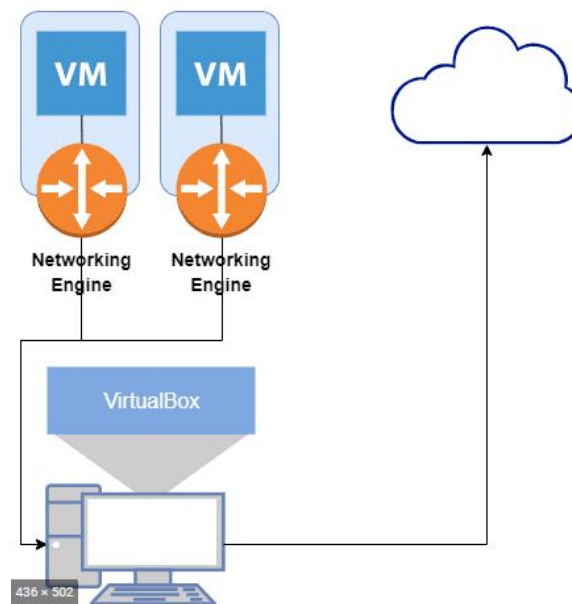
- Gestionar las solicitudes de un gran número de usuarios. Se trata de un problema de escalabilidad que surge con el continuo crecimiento del número de usuarios activos en el sistema.
- Cuando las páginas tardan mucho en cargar, la experiencia del usuario puede ser molesta y frustrante.
- Flujo masivo de datos hace que los recursos del servidor no sean suficientes, lo que provoca que colapse y deje de funcionar.

Solución implementada.



Clúster de servidores web

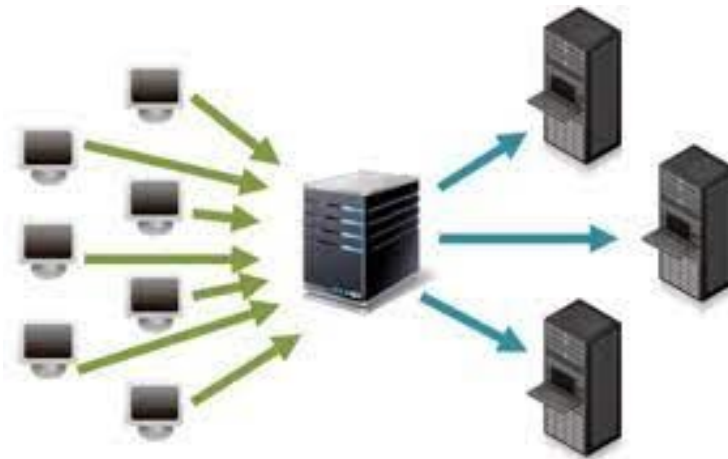
- Un clúster es básicamente, un sistema informático donde varios servidores se unen para funcionar como si fuesen uno solo.
- Un clúster de balanceo de carga está compuesto por uno o más ordenadores (llamados nodos) que actúan como frontend (Interfaz de usuario) del clúster, y que se ocupan de repartir las peticiones de servicio que reciba el clúster, a otros ordenadores del clúster que forman el back-end (Servidor) de éste.



Balanceador de carga

¿Qué es? ¿Por qué es importante?

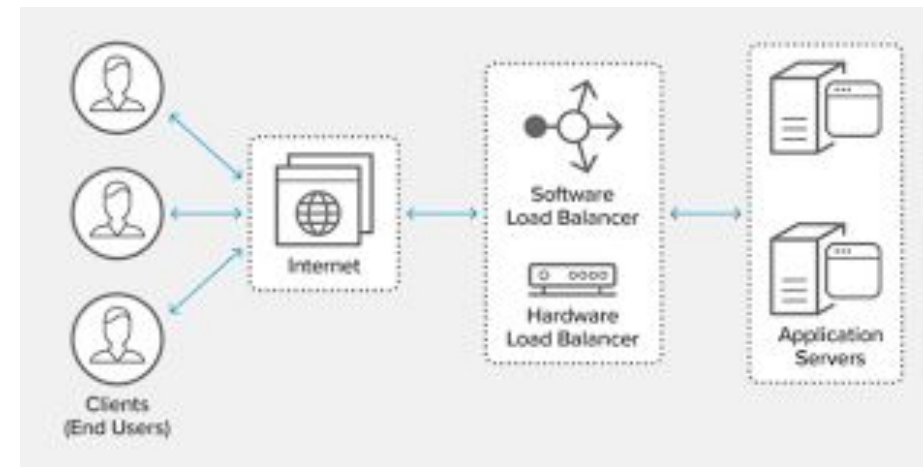
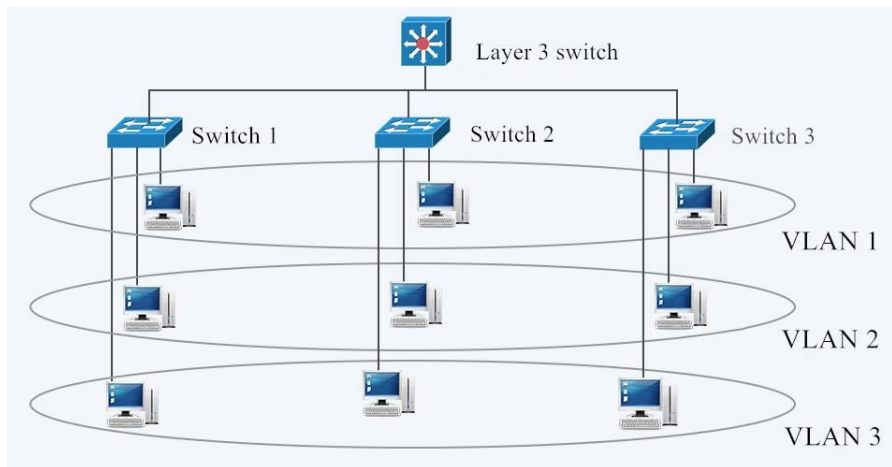
- Mediante el balanceo, todas las solicitudes al sistema se reparten entre diversos servidores y, con ello, se aumenta la cantidad de usuarios concurrentes que un proyecto puede atender.
- Gestión del tráfico. Es una manera ágil de gestionar las solicitudes entrantes y salientes de datos.



Tipos de balanceadores de carga

Fundamentalmente se clasifican en tres tipos:

- De tipo hardware.
- Basado en software.



Balanceadores de carga tipo software

- HAProxy.
- Pen.
- Nginx.
- Etc



Métodos de balanceo de carga

- Round Robin.
- Least Connected.
- IP Hash.

Implementación

- Crear clúster con vagrant y virtualBox.
- Iniciar las máquinas.
- Instalar servicios(nginx,bind,EPEL,DNS).
- Configuración DNS(archivos de zona).
- Configuración nginx.
- Pruebas.

Funcionamiento.

The screenshot displays the OBS Studio interface with two windows open. The main window shows a terminal with the configuration for an Nginx load balancer. The Notepad window shows the Vagrantfile configuration for three virtual machines: nginx, backend1, and backend2.

```

root@nginx/etc/nginx/conf.d
root@backend1/usr/share/nginx
root@backend2/usr/share/nginx

upstream backend {
    ip_hash;
    server 192.168.100.3;
    server 192.168.50.4;
}

server {
    listen 80;
    server_name server.pjfinal.com;
    server_name www.pjfinal.com;
    server_name 192.168.100.2;

    location / {
        proxy_redirect off;
        proxy_set_header    X-Real-IP $remote_addr;
        proxy_set_header    X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header    Host $http_host;
        proxy_pass http://backend;
    }
}

~
~
~
~

No source selected
Scene
Scene
"load-balancing.conf" 20L, 401C

+ - ^ v
+ - ^ v

Mic/Aux
0.0 dB

LIVE: 00:00:00 REC: 00:01:48 CPU: 3.8%, 60.00 fps
Type here to search
943 p.m. 6/11/2021
  
```

```

*Vagrantfile - Notepad
File Edit Format View Help
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure("2") do |config|
  if Vagrant.has_plugin? "vagrant-vbguest"
    config.vbguest.no_install = true
    config.vbguest.auto_update = false
    config.vbguest.no_remote = true
  end
  config.vm.define :nginx do |nginx|
    nginx.vm.box = "bento/centos-7.9"
    nginx.vm.network :private_network, ip: "192.168.100.2"
    nginx.vm.hostname = "nginx"
  end
  config.vm.define :backend1 do |backend1|
    backend1.vm.box = "bento/centos-7.9"
    backend1.vm.network :private_network, ip: "192.168.100.3"
    backend1.vm.hostname = "backend1"
  end
  config.vm.define :backend2 do |backend2|
    backend2.vm.box = "bento/centos-7.9"
    backend2.vm.network :private_network, ip: "192.168.50.4"
    backend2.vm.hostname = "backend2"
  end
end
  
```

Conclusiones.

- Para la elaboración de este trabajo se indagaron de manera proporcional las herramientas que permiten configurar un balanceo de carga de manera satisfactoria, aunque se eligió NGINX por su facilidad de implementación, además de ser de código abierto y consumir menos recursos.
- A pesar de las herramientas mostradas durante este proyecto, es importante mencionar que existen en el mercado herramientas que permiten realizar el balanceo de carga junto a muchas otras operaciones de manera automatizada, estas son plataformas o servicios como AWS, Azure, entre otros.
- Implementar un balanceador de carga por medio de software tiene la ventaja de ser muy económico, relativamente sencillo y fácil de usar.



Universidad Autónoma de Occidente - Cali

Gracias.

