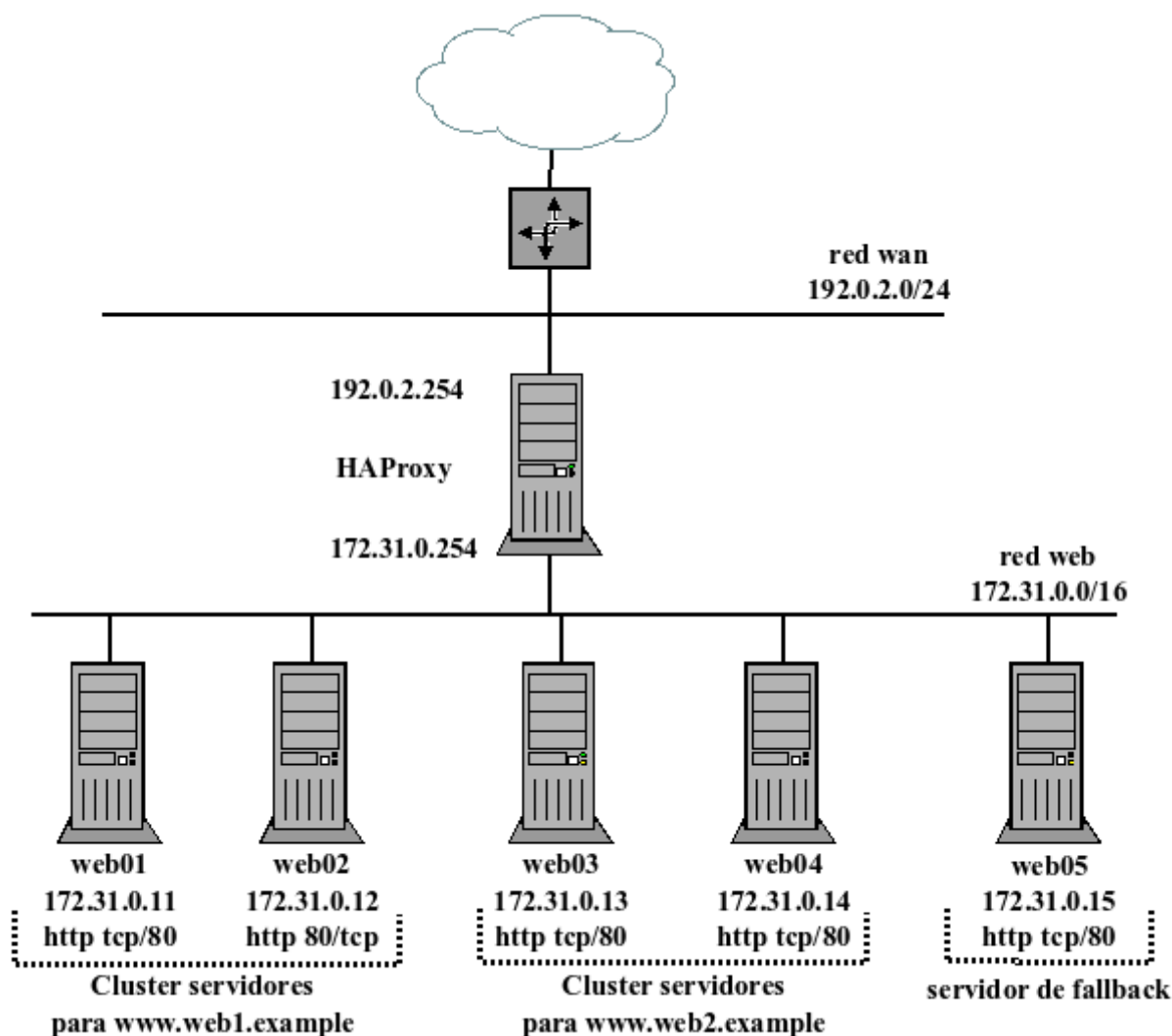


Usaremos HAProxy para dar servicio a la siguiente infraestructura con los siguientes requisitos:



HAPROXY

- Balancear los servidores del cluster de `www.web1.example` con RoundRobin y los de `www.web2.example` con RoundRobin con peso, asignándole a web03 un peso 3 y a web04 un peso de 1.
- Comprobar el estado de los servidores web verificando que el contenido del fichero `test.html` sea OK.
- Habilitar el sitio web de estadísticas de HAProxy en la IP interna puerto `tcp/8180` protegiendo el acceso para el usuario administrador con la contraseña `abc123`. Y refrescando la página cada 10 segundos.

NOTA:

- Aprovecha el escenario resultante de la práctica guiada nº1.
- Los servidores web ya tienen un fichero `test.html` con OK en la ubicación `/var/www/localhost/htdocs/test.html` que es el DocumentRoot del servidor web (carpeta donde va a buscar los ficheros el servidor web y que se corresponde con la raíz en las peticiones de los cliente).
- Los servidores web corren Alpine Linux y se puede acceder a ellos mediante el comando `lxc exec web0X ash`

A entregar:

- Fichero de configuración de HAProxy.

- Vídeo 1: debe dividirse la pantalla en dos para ver:
 - Salida en tiempo real del fichero de log de HAProxy.
 - Terminal de un cliente desde donde se ejecutará:
 - 4 veces seguidas el comando: `curl http://www.web1.example`
 - a continuación 5 veces seguidas el comando: `curl http://www.web2.example`

NOTAS:

- el objetivo del vídeo es ver como se produce el balanceo por lo que es importante que sea legible tanto la salida del log de HAProxy como la página web devuelta.
- el cliente puede ser el equipo anfitrión de los contenedores
- Vídeo 2: debe dividirse la pantalla en tres:
 - Salida en tiempo real del fichero de log de HAProxy.
 - Terminal de un cliente que solicitará páginas web a HAProxy.
 - Terminal de web01.

Ejecutar la siguiente secuencia de acciones:

- ejecutar 2 veces seguidas el comando: `curl http://www.web1.example`
- en web01 editar el fichero `test.html` que está en `/var/www/localhost/htdocs/test.html` y poner su valor a OFF
- esperar a que HAProxy muestre los mensajes de error conforme web01 está caído
- ejecutar 2 veces seguidas el comando: `curl http://www.web1.example`
- en web01 editar el fichero `test.html` y poner su valor nuevamente a OK
- esperar a que HAProxy muestre los mensajes conforme web01 está operativo de nuevo
- ejecutar 2 veces seguidas el comando: `curl http://www.web1.example`

NOTAS:

- el objetivo del vídeo es ver como se produce HAProxy chequea el estado de los servidores del cluster.
- Para entrar en web01 hay que ejecutar: `lxc exec web01 ash`
- Una vez dentro de web01, se puede usar el editor `nano` para cambiar el contenido de `test.html`
- Captura de pantalla de la página de estadísticas de HAProxy después de ejecutar los siguientes comandos:
 - `for i in {1..100} ; do curl http://www.web1.example ; done`
 - `for i in {1..400} ; do curl http://www.web2.example ; done`