QUESTION:

Set up a cluster of Linux servers running a web server (e.g., Apache) with a load balancer (e.g., HAProxy) for redundancy and scalability.

Demonstrate expertise in system administration, high availability (HA) concepts, load balancing, and web server configuration.

ANSWER:

PREREQUISITES:

Choose your Linux Distribution: Select a suitable Linux distribution for your servers (e.g., Ubuntu, CentOS). Ensure all servers use the same distribution.

Server Configuration: Make sure all servers (3 Servers) have the same hardware specifications and network connectivity.

Domain Name and IP Addresses: Have a registered domain name and static IP addresses for each server and the load balancer.

e.g.,

3 Ubuntu distribution 22.04/24.04 VPS(Virtual Private Server)

-VPS 1: Install HAProxy

IP1:172.31.89.194

root\_password: Ex@123

-VPS 2,3: Use as a web Server

IP2:172.31.92.29

root\_password:Ex@123

IP3:172.31.95.220

root\_password:Ex@123

To change hostname on each server:

Haproxy:

hostnamectl set-hostname proxy .........( to set permanent hostname)

hostnamectl set-hostname Webserver1 .......(to set permanent hostname)

hostnamectl set-hostname Webserver2.....(to set permanent hostname)

###########################################################################

configuration in /etc/Hosts file

add ip address with hostname name of alternate server to ping each other

on all server

e.g.,



-VPS 1: HAProxy(Loadbalancer)

<INSTALL HAPROXY>

$sudo su ..........( Switch to root user/access )

#lsb\_release -a ..........( to describe version of you distribution VPS)

e.g.,

Output:

No LSB modules are available

distributor ID: Ubuntu

Description: Ubuntu 22.04.1 LTS

Release: 22.04\

codename: example

#apt update -y && apt upgrade -y ..........( The first command updates the package lists, while the second upgrades the installed packages to their latest versions)

#apt install HAProxy -y .........( Install HAProxy for Loadbalancer)

#apt show HAProxy ...........( HAProxy has always three active stable versions of the releases, two of the latest versions in development plus a third older version that is still receiving critical updates. You can always check the currently newest stable version listed on the HAProxy website and then decide which version you wish to go with.)

e.g

Output

package: HAProxy

Version: 2.4.22-0Ubuntu0.22.04.1

priority : Optional

Section: net

Origin:Ubuntu

etc.,

#apt install Software-properties-common ........(provides an abstraction of the used apt repositories. It allows you to easily manage your distribution and independent software vendor software sources)

#add-apt-repository ppa:vbernat/haproxy-2.8 ......(While the latest stable version 2.8 of HAProxy is not yet available on the packet manager by default, it can be found in a third-party repository. To install HAProxy from an outside repo, you will need to add the new repository with the following command) Confirm adding the new PPA by pressing the Enter key.

#apt update

#HAProxy -v ( To check latest upgraded version of HAProxy for loadbalancer )

e.g.,

Output:

HA-Proxy version 2.8.8-1ppa1~xenial 2017/07/09

Copyright 2000-2017 Willy Tarreau <willy@haproxy.org>

<CONFIGURE HAPROXY>

#cp -a /etc/haproxy/haproxy.cfg{.,orig}

#apt install vim -y

#vim /etc/haproxy/haproxy.cfg

e.g.,

Output:

global

log /dev/log local0

log /dev/log local1 notice

chroot /var/lib/haproxy

stats socket /run/haproxy/admin.sock mode 660 level admin

stats timeout 30s

user haproxy

group haproxy

daemon

# Default SSL material locations

ca-base /etc/ssl/certs

crt-base /etc/ssl/private

# See: <https://ssl-config.mozilla.org/#server=haproxy&server-version=2.0.3&config=intermediate>

ssl-default-bind-ciphers ECDHE-ECDSA-AES128-GCM-SHA256:ECDHE-RSA-AES128-GCM-SHA256:ECDHE-ECDSA-AES256-GCM-SHA384:ECDHE-RSA-AES256-GCM-SHA384:ECDHE-ECDSA-CHACHA20-POLY1305:ECDHE-RSA-CHACHA20-POLY1305:DHE-RSA-AES128-GCM-SHA256:DHE-RSA-AES256-GCM-SHA384

ssl-default-bind-ciphersuites TLS\_AES\_128\_GCM\_SHA256:TLS\_AES\_256\_GCM\_SHA384:TLS\_CHACHA20\_POLY1305\_SHA256

ssl-default-bind-options ssl-min-ver TLSv1.2 no-tls-tickets

defaults

log global

mode http

option httplog

option dontlognull

timeout connect 5000

timeout client 50000

timeout server 50000

errorfile 400 /etc/haproxy/errors/400.http

errorfile 403 /etc/haproxy/errors/403.http

errorfile 408 /etc/haproxy/errors/408.http

errorfile 500 /etc/haproxy/errors/500.http

errorfile 502 /etc/haproxy/errors/502.http

errorfile 503 /etc/haproxy/errors/503.http

errorfile 504 /etc/haproxy/errors/504.http

#####manually add below configuration command into HA-Proxy Configuration File #####

frontend haproxy-main

bind \*:80

option forwardfor

default\_backend apache\_webservers

backend apache\_webservers

balance roundrobin

server WS1 172.31.92.29:80 check or server WS1 172.31.92.29 check port 80

server WS2 172.31.95.220:80 check or server WS2 172.31.95.220 check port 80

listen stats

bind :8800

stats enable

stats uri /

stats hide-version

stats auth username:password

default\_backend apache\_webservers

save file: Esc :wq!

###########################################################################

OR

#####manually add below configuration command into HA-Proxy Configuration File #####

#sudo nano /etc/haproxy/haproxy.cfg

frontend http\_front

bind \*:80

stats uri /haproxy?stats

default\_backend http\_back

backend http\_back

balance roundrobin

server <server1 name> <private IP 1>:80 check

server <server2 name> <private IP 2>:80 check

###########################################################################

#systemctl restart service

Testing the Setup

[http://<load](http://%3Cload/) balancer public IP>/haproxy?stats



To Verify and check status of Haproxy

#sudo systemctl status haproxy

**Password protecting the statistics page (Optional)**

However, Having the statistics page listed at the front end is publicly open for anyone to view, which might not be such a good idea. Instead, you can set it up to its own port number by adding the example below to the end of your haproxy.cfg file. Replace the username and password with something secure.

#vim /etc/haproxy/haproxy.cfg

listen stats

bind \*:8181

stats enable

stats uri /

stats realm Haproxy Statistics

stats auth username:password

##After adding the new listen group, remove the old reference to the stats uri from the frontend group. When done, save the file and restart HAProxy again.##

#sudo systemctl restart haproxy

Then open the load balancer again with the new port number, and log in with the username and password you set in the configuration file.

[http://<load](http://%3Cload/) balancer public IP>:8181

Check that your servers are still reporting all green, and then open just the load balancer IP without any port numbers on your web browser.

[http://<load](http://%3Cload/) balancer public IP>/

output

===========================================================================

-VPS 2 - Apache2 installation and Hosting

$sudo su

#apt update -y && apt upgrade -y

#apt install apache2 -y

#systemctl enable apache2

#systemctl start apache2

#echo "Welcome From Webserver1 server running fine" | sudo tee /var/www/html/index.html

output:"Welcome From Webserver1 server running fine"

For Testing: chrome --> http:// your\_ipaddress

output:Welcome From Webserver1 server running fine

-VPS 3 - Apache2 installation and Hosting

$sudo su

#apt update -y && apt upgrade -y

#apt install apache2 -y

#systemctl enable apache2

#systemctl start apache2

#echo "Welcome From Webserver2 server running fine" | sudo tee /var/www/html/index.html

For Testing: chrome --> http:// your\_ipaddress

output:Welcome From Webserver2 server running fine

output:"Welcome From Webserver2 server running fine"

or

while true; do curl localhost; sleep 1; done