**Chương V : Cài đặt và cấu hình hệ thống**

**1. Cài đặt Centos 6.7 và các gói trên server 1 , 2 , 3 và 4**

Cài đặt các gói :

sudo yum update

sudo yum install epel-release

sudo yum install gcc-c++ -y

**2. Compile Nginx from source on Centos 6.7**

**Step 1 :**

Tải source Nginx về :

sudo wget http://nginx.org/download/nginx-1.6.0.tar.gz

Giải nén

sudo tar xfz nginx-1.6.0.tar.gz

sudo cd nginx-1.6.0/

./configure -- user=nginx -- group=nginx -- prefix=/etc/nginx -- sbin-path=/usr/sbin/nginx --

conf-path=/etc/nginx/nginx.conf -- error-log- path=/var/log/nginx/error.log -- http-log-

path=/var/log/nginx/access.log -- pid-path=/var/run/nginx.pid -- lock-path=/var/run/nginx.lock

-- with-http\_ssl\_module -- with-pcre

**Xem thông tin cấu hình nginx qua câu lệch bên dưới**

./configure -- help

user=nginx –group=nginx # khai bao user va group nginx

prefix=/etc/nginx # Đường dẫn nginx (nginx.conf)

sbin-path=/usr/sbin/nginx # Đường dẫn thực thị file nginx

error-log- path=/var/log/nginx/error.log # log nginx

pid-path=/var/run/nginx.pid # name process ID

sudo make && make install

**Step 2 : Cấu hình file nginx.conf**

Tạo user nginx

useradd nginx

Vào file cấu hình nginx /etc/nginx/nginx.conf thay đổi dòng

sudo vi /etc/nginx/nginx.conf

#user nologin ===&gt; user nginx;

Sau khi thay đổi hết những thông số trên kiểm tra xem Nginx có đang chạy hay không :

netstat -tulpn | grep nginx

tcp 0 0 0.0.0.0:80 0.0.0.0:\* LISTEN 11803/nginx

**Chúng ta tạo 1 file thực thi để quản lý nginx trong /etc/init.d/nginx**

sudo vi /etc/init.d/nginx

**Paste bash này vào :**

#!/bin/sh

# chkconfig: - 85 15

# description: Nginx is an HTTP(S) server, HTTP(S) reverse \

# proxy and IMAP/POP3 proxy server

# processname: nginx

# config: /etc/nginx/nginx.conf

# pidfile: /var/run/nginx.pid

# user: nginx

# Source function library.

. /etc/rc.d/init.d/functions

# Source networking configuration.

. /etc/sysconfig/network

# Check that networking is up.

[ &quot;$NETWORKING&quot; = &quot;no&quot; ] &amp;&amp; exit 0

nginx=&quot;/usr/sbin/nginx&quot;

prog=$(basename $nginx)

NGINX\_CONF\_FILE=&quot;/etc/nginx/nginx.conf&quot;

lockfile=/var/run/nginx.lock

start() {

[ -x $nginx ] || exit 5

[ -f $NGINX\_CONF\_FILE ] || exit 6

echo -n $&quot;Starting $prog: &quot;

daemon $nginx -c $NGINX\_CONF\_FILE

retval=$?

echo

[ $retval -eq 0 ] &amp;&amp; touch $lockfile

return $retval

}

stop() {

echo -n $&quot;Stopping $prog: &quot;

killproc $prog -QUIT

retval=$?

echo

[ $retval -eq 0 ] &amp;&amp; rm -f $lockfile

return $retval

}

restart() {

configtest || return $?

stop

start

}

reload() {

configtest || return $?

echo -n $&quot;Reloading $prog: &quot;

killproc $nginx -HUP

RETVAL=$?

echo

}

force\_reload() {

restart

}

configtest() {

$nginx -t -c $NGINX\_CONF\_FILE

}

rh\_status() {

status $prog

}

rh\_status\_q() {

rh\_status &gt;/dev/null 2&gt;&amp;1

}

case &quot;$1&quot; in

start)

rh\_status\_q &amp;&amp; exit 0

$1

;;

stop)

rh\_status\_q || exit 0

$1

;;

restart|configtest)

$1

;;

reload)

rh\_status\_q || exit 7

$1

;;

force-reload)

force\_reload

;;

status)

rh\_status

;;

condrestart|try-restart)

rh\_status\_q || exit 0

;;

\*)

echo $&quot;Usage: $0 {start|stop|status|restart|condrestart|try-restart|reload|force-

reload|configtest}&quot;

exit 2

esac

**Cấp quyền thực thi cho file :**

sudo chmod +x /etc/init.d/nginx

**Start Nginx :**

sudo chkconfig nginx on

sudo /etc/init.d/nginx start

**3.Cấu hình Virtual Host :**

**Set up Nginx server blocks (Virtual Hosts) on Centos**

**Step 1 :**

Tạo đường dẫn html

mkdir -p /var/www/nhom2.com/html

Chuyển giao quyền sở hữu cho người sử dụng thường xuyên. Biến môi trường ở

đây là $USER để thay thế các tài khoản người dùng đăng nhập vào. Điều này cho phép. Chúng ta tạo ra các tập tin trong thư mục này mà không cho phép người khác chỉnh sửa hay tạo ra nội dung :

chown -R $USER:$USER /var/www/nhom2.com/html

Kiểm tra để chắc chắn rằng giá trị umask không thay đổi

chmod -R 755 /var/www

**Step 2 :**

Tạo ra file index.html trong thư mục chứa tên miền đầu tiên :

vi /var/www/nhom2.com/html/index.html

<html>

<head>

<title>Welcome to nhom2.com !</title>

</head>

<body>

<h1>Success! The nhom2.com server block is working!</h1>

</body>

</html>

**Step 3 :**

Tạo thư mục để tạo file cấu hình cho domain :

sudo mkdir /etc/nginx/sites-available

sudo mkdir /etc/nginx/sites-enabled

**Step 4 :**

sudo vi /etc/nginx/nginx.conf

add line http{} :

include /etc/nginx/sites-enabled/\*.conf;

server\_names\_hash\_bucket\_size 64;

**Step 5 :**

sudo vi /etc/nginx/sites-available/nhom2.com.conf

server {

listen 80;

server\_name nhom2.com www.nhom2.com;

location / {

root /var/www/nhom2.com/html;

index index.html index.htm;

try\_files $uri $uri/ =404;

}

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root /usr/share/nginx/html;

}

}

sudo ln -s /etc/nginx/sites-available/quangthoai.com.conf /etc/nginx/sites-

enabled/nhom2.com.conf

**Step 6 : Add host :**

sudo nano /etc/hosts

127.0.0.1 localhost

127.0.1.1 guest-desktop

192.168.244.150 nhom2.com

**Step 7: Cấu hình trong file /etc/nginx/nginx.config để loadbalancing**

worker\_processes 1;

events {

worker\_connections 1024;

}

# loabalancing dùng giải thuật round robin request đá về server 141 và 142 theo tỷ lệ 3 : 1

http {

upstream servers {

server 192.168.244.141 weight=3;

server 192.168.244.142;

}

server {

listen 80;

server\_name nhom2.com www.nhom2.com;

location / {

root /var/www/nhom2.com/html;

index index.html index.htm;

proxy\_pass http://servers;

try\_files $uri $uri/ =404;

}

error\_page 500 502 503 504 /50x.html;

location = /50x.html {

root /usr/share/nginx/html;

}

**5. Cấu hình firewall cho server 1 và server 4**

Trên server 1 ( 192.168.244.140) ,

Cho phép VRRP traffic và cấu hình firewall:

# sudo iptables -I INPUT -p vrrp -m comment --comment "VRRP" -j ACCEPT

# sudo service iptables save

Để cho dịch vụ keepalived chuyển tiếp gói tin đến các máy chủ thật sự , nỗi node của server chuyển tiếp bật trong kernel :

# sudo sed -i 's/net.ipv4.ip\_forward = 0/net.ipv4.ip\_forward = 1/' /etc/sysctl.conf

# sudo sysctl -p

Cho phép chạy loadbalancer khi server failover :

# sudo echo "net.ipv4.ip\_nonlocal\_bind=1" >>/etc/sysctl.conf

# sudo sysctl -p

Copy firewall từ server 1 sang server 4 :

# sudo scp /etc/sysconfig/iptables root@server4:/etc/sysconfig/

# sudo scp /etc/sysctl.conf root@s:/etc/

**6. Cài đặt và cấu hình Keepalived:**

**Step 1** :

Cài đặt keepalived trên server 1 và server 4

với command :

sudo yum install keepalived

**Step 2**:

Cấu hình :

**Trên server 1 :**

vi /etc/keepalived/keepalived.conf

#Script used to check if nginx is running

vrrp\_script check\_nginx {

script "killall -0 nginx "

interval 2

weight 2

}

vrrp\_instance VI\_1 {

state MASTER

interface eth0

virtual\_router\_id 51

priority 100

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

192.168.244.150

}

track\_script {

check\_nginx

}

}

**Trên server 4** :

vi /etc/keepalived/keepalived.conf

#Script used to check if nginx is running

vrrp\_script check\_nginx {

script "killall -0 nginx "

interval 2

weight 2

}

vrrp\_instance VI\_1 {

state SLAVE

interface eth0

virtual\_router\_id 51

priority 99

advert\_int 1

authentication {

auth\_type PASS

auth\_pass 1111

}

virtual\_ipaddress {

192.168.244.150

}

track\_script {

check\_nginx

}

}

**Step 3 : Start service keepalived và tiến hành test thử :**

sudo /etc/init.d/keepalived start