**BÁO CÁO BÀI KIỂM TRA**

*Nguyễn Ngọc Minh Trí*

**MỤC LỤC**

[**MỤC LỤC** 2](#_Toc195365818)

[1. Triển khai hệ thống webserver nginx trên 2 server Web01 và Web02. Đảm bảo source code khi upload lên web01 thì web02 sẽ được đồng bộ theo bằng lsyncd 3](#_Toc195365819)

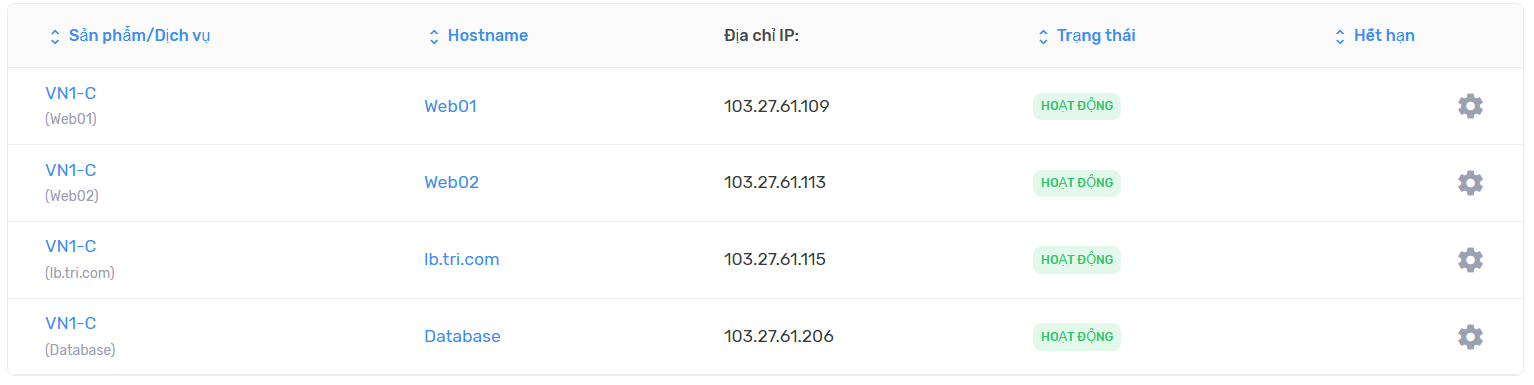
[2. Triển khai hệ thống load balancing bằng haproxy (load balancer) trên Load Balancer với cơ chế roundrobin và healthcheck đảm bảo khi có bất kỳ server nào gặp sự cố thì phát hiện ngay và ngưng chuyển request vè webserver đó 8](#_Toc195365820)

[3. Triển khai cluster database MySQL Galera trên 3 server web01, web02, và Database Server. Sau đó sử dụng Load Balancer ở câu 2 để cấu hình và điều phối traffic cho bên ngoài kết nối vào database 12](#_Toc195365821)

[4. Tạo database và cài đặt 1 website wordpress đơn giản với database vừa tạo. Phân quyền cho database chỉ cho 5 địa chỉ IP sau truy cập 19](#_Toc195365822)

## Triển khai hệ thống webserver nginx trên 2 server Web01 và Web02. Đảm bảo source code khi upload lên web01 thì web02 sẽ được đồng bộ theo bằng lsyncd

|  |  |
| --- | --- |
| Load Balancer | 192.168.115.1 |
| Web01 | 192.168.115.2 |
| Web02 | 192.168.115.3 |
| Database Server | 192.168.115.4 |



**Gắn card mạng:**

Trên Load Balancer:

**ip addr add 192.168.115.1/24 dev ens19**

**ip link set ens19 up**

Trên Web01:

**ip addr add 192.168.115.2/24 dev ens19**

**ip link set ens19 up**

Trên Web02:

**ip addr add 192.168.115.3/24 dev ens19**

**ip link set ens19 up**

Trên Database Server:

**ip addr add 192.168.115.4/24 dev ens19**

**ip link set ens19 up**

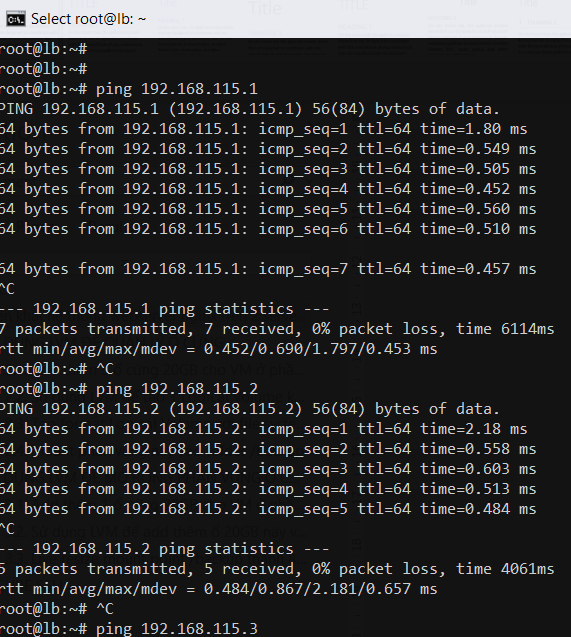
Ping test nhanh - từ mỗi máy, thử:

**ping 192.168.115.1**

**ping 192.168.115.2**

**ping 192.168.115.3**

**ping 192.168.115.4**



**Cài đặt nginx:**

Bước 1: Cài đặt nginx trên Web01 và Web02

**sudo apt update**

**sudo apt install nginx -y**

Bước 2: Cấu hình thư mục chứa mã nguồn website

Giả sử mã nguồn web nằm trong /var/www/html/

**sudo mkdir -p /var/www/html**

**sudo chown -R www-data:www-data /var/www/html**

Bước 3: Thiết lập SSH không cần mật khẩu từ Web01 → Web02

Trên Web01:

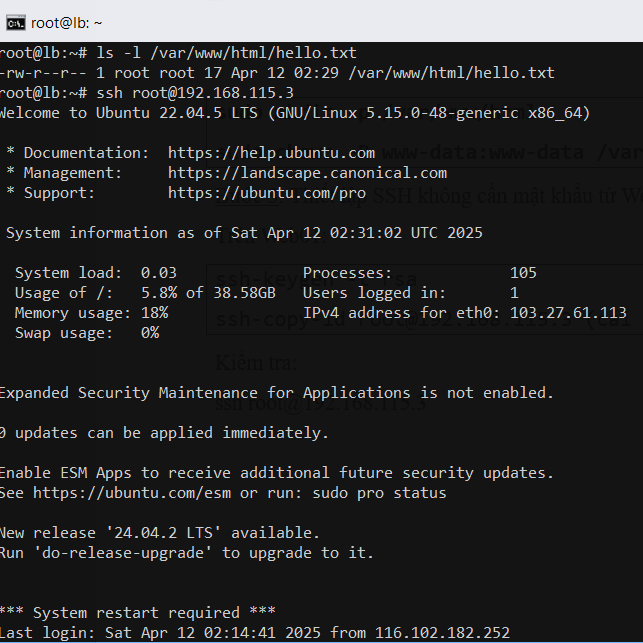
**ssh-keygen -t rsa**

**ssh-copy-id root@192.168.115.3**

(cái IP của Web02)

Kiểm tra:

**ssh root@192.168.115.3**



**Cài đặt lsyncd (Live Syncing Daemon) trên Web01:**

Cơ chế hoạt động của lsyncd

* Web01 là máy chủ chứa source và Web02 là máy được đồng bộ
* lsyncd chạy trên Web01
* Nó theo dõi thay đổi trong thư mục source (/var/www/html, chẳng hạn)
* Khi có thay đổi, nó sử dụng rsync qua SSH để đẩy file sang Web02 (Nó hoạt động theo cơ chế master - slave)

**Trên máy Web01**

Chạy lệnh để cài đặt lsyncd:

**sudo apt install lsyncd -y**

Tạo file cấu hình /etc/lsyncd/lsyncd.conf.lua

**sudo mkdir -p /etc/lsyncd**

**sudo nano /etc/lsyncd/lsyncd.conf.lua**

Thêm nội dung sau:

settings {

logfile = "/var/log/lsyncd.log",

statusFile = "/var/log/lsyncd.status",

statusInterval = 20

}

sync {

default.rsyncssh,

source = "/var/www/html",

host = "root@192.168.115.3",

targetdir = "/var/www/html",

rsync = {

archive = true,

compress = true,

verbose = true

}

}

Khởi động lsyncd:

**sudo systemctl enable lsyncd**

**sudo systemctl start lsyncd**

Xem log nếu muốn kiểm tra:

**tail -f /var/log/lsyncd.log**

**Trên máy Web02**

Đảm bảo thư mục /var/www/html tồn tại và có quyền ghi

**mkdir -p /var/www/html**

**chmod -R 755 /var/www/html**

**chown -R root:root /var/www/html**

**Test xem lsyncd đã đồng bộ được chưa:**

Trên Web01 - Tạo 1 file test ở thư mục source:

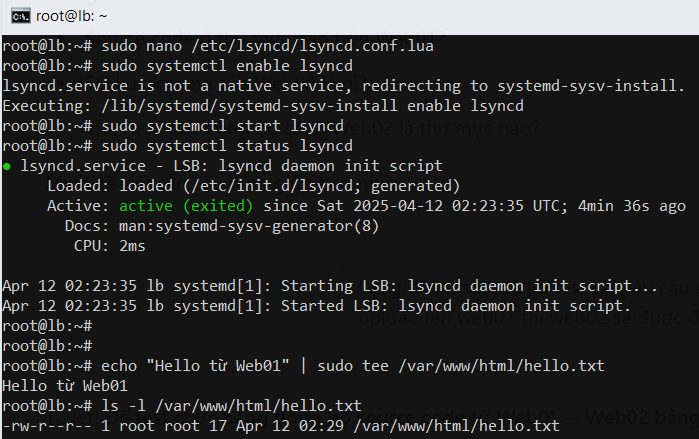
**echo "Hello từ Web01" | sudo tee /var/www/html/hello.txt**

Trên Web02 - Kiểm tra xem file hello.txt đã xuất hiện chưa:

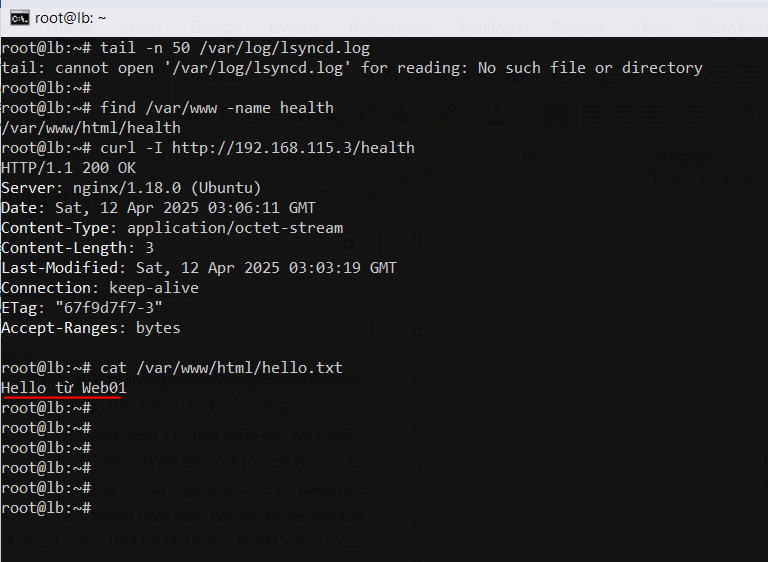
**cat /var/www/html/hello.txt**

Nếu kết quả là "Hello từ Web01" → Đồng bộ thành công

**MÁY Web01**

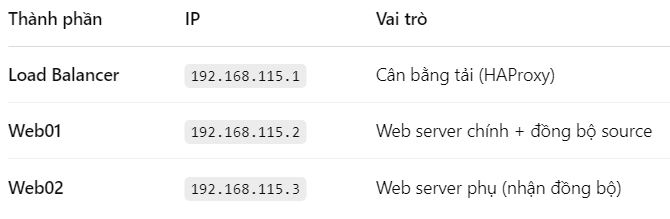


**MÁY Web02**



## Triển khai hệ thống load balancing bằng haproxy (load balancer) trên Load Balancer với cơ chế roundrobin và healthcheck đảm bảo khi có bất kỳ server nào gặp sự cố thì phát hiện ngay và ngưng chuyển request vè webserver đó

*Máy này sẽ đóng vai trò phân phối request đến Web01 và Web02.*



Bước 1: Cài đặt HAProxy trên Load Balancer:

**sudo apt update**

**sudo apt install haproxy**

Khởi động HAProxy

**sudo systemctl start haproxy**

**sudo systemctl enable haproxy**

Bước 2: Cấu hình HAProxy

Chỉnh sửa file cấu hình HAProxy (/etc/haproxy/haproxy.cfg):

**sudo nano /etc/haproxy/haproxy.cfg**

Cấu hình load balancing với roundrobin và health check:

global

log /dev/log local0

log /dev/log local1 notice

chroot /var/lib/haproxy

stats timeout 30s

user haproxy

group haproxy

daemon

defaults

log global

option httplog

option dontlognull

timeout connect 5000ms

timeout client 50000ms

timeout server 50000ms

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

frontend http\_front

bind \*:80

default\_backend web\_servers

backend web\_servers

balance roundrobin

option httpchk GET /health

http-check expect status 200

server web01 192.168.115.2:80 check

server web02 192.168.115.3:80 check

listen stats

bind \*:8080

stats enable

stats uri /stats

stats realm Haproxy\ Statistics

stats auth admin:123456

Giải thích cấu hình:

* global và defaults: Cấu hình toàn cục cho HAProxy.
* frontend http\_front: Định nghĩa cổng và giao thức vào (ở đây là cổng 80).
* backend http\_back: Cấu hình các server web phía sau load balancer. Sử dụng balance roundrobin để phân phối đều các yêu cầu đến Web01 và Web02.
* option httpchk GET /health: HAProxy sẽ gửi một yêu cầu HTTP đến /health trên các server web để kiểm tra tình trạng.
* server web01 và server web02: Định nghĩa các server web, với tùy chọn check để bật health check.

Cấu hình health check: Đảm bảo rằng có một endpoint /health trên cả Web01 và Web02 để HAProxy có thể thực hiện kiểm tra sức khỏe. Ví dụ, có thể tạo một trang đơn giản trong Nginx để trả về mã HTTP 200 nếu server hoạt động bình thường.

Trong file cấu hình Nginx của Web01 và Web02 (/etc/nginx/sites-available/default), thêm một location cho /health:

server {

listen 80;

server\_name localhost;

location / {

root /var/www/html;

index index.html index.htm;

}

location /health {

return 200 'OK';

}

}

Sau đó, reload lại Nginx:

**sudo systemctl reload nginx**

Bước 3: Kiểm tra và khởi động lại HAProxy

Kiểm tra cấu hình HAProxy: Trước khi khởi động lại HAProxy, hãy kiểm tra cấu hình

**sudo haproxy -c -f /etc/haproxy/haproxy.cfg**

Nếu không có lỗi sẽ nhận được thông báo Configuration file is valid.

Khởi động lại HAProxy để áp dụng cấu hình mới:

**sudo systemctl restart haproxy**

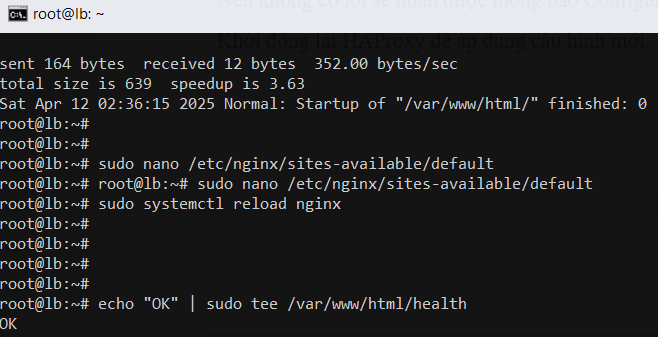
**Ở máy Web01:** Tạo file /health trên Web01

**echo "OK" | sudo tee /var/www/html/health**

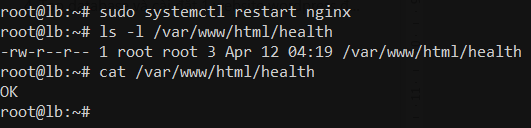
Do bạn đã cấu hình lsyncd, file này sẽ tự đồng bộ sang Web02. Kiểm tra lại ở Web02:

cat /var/www/html/health

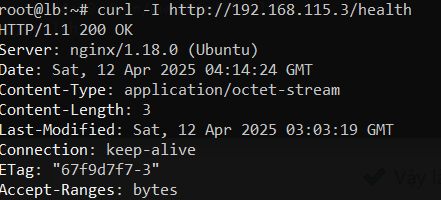
**MÁY Web01:**



**MÁY Web02:**

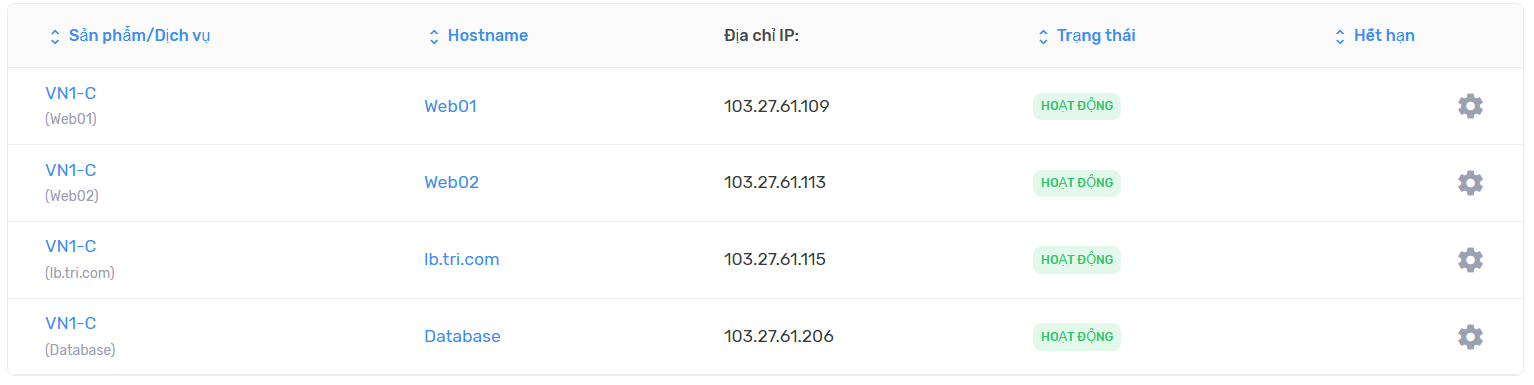


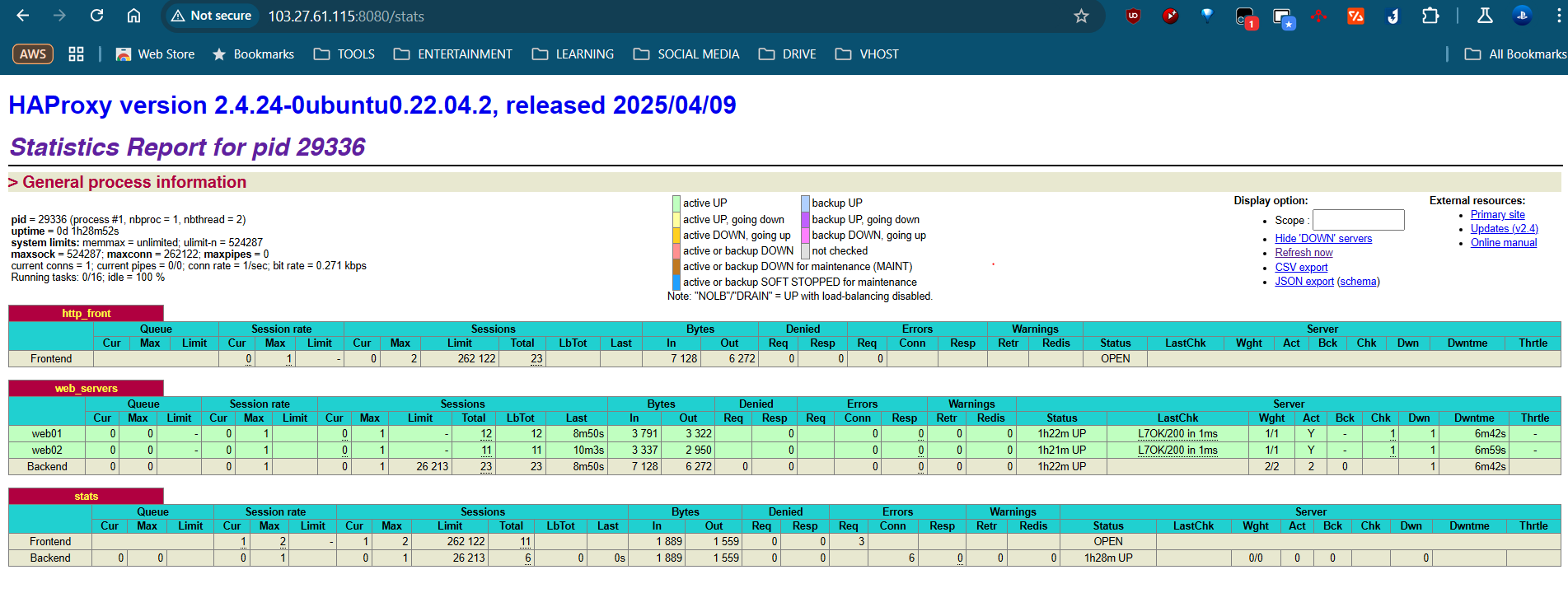
**CHECK LẠI DÙNG LỆNH CURL TRÊN CẢ 3 MÁY:**

d

**XEM THỐNG KÊ WEB UI:**

103.27.61.115:8080/stats





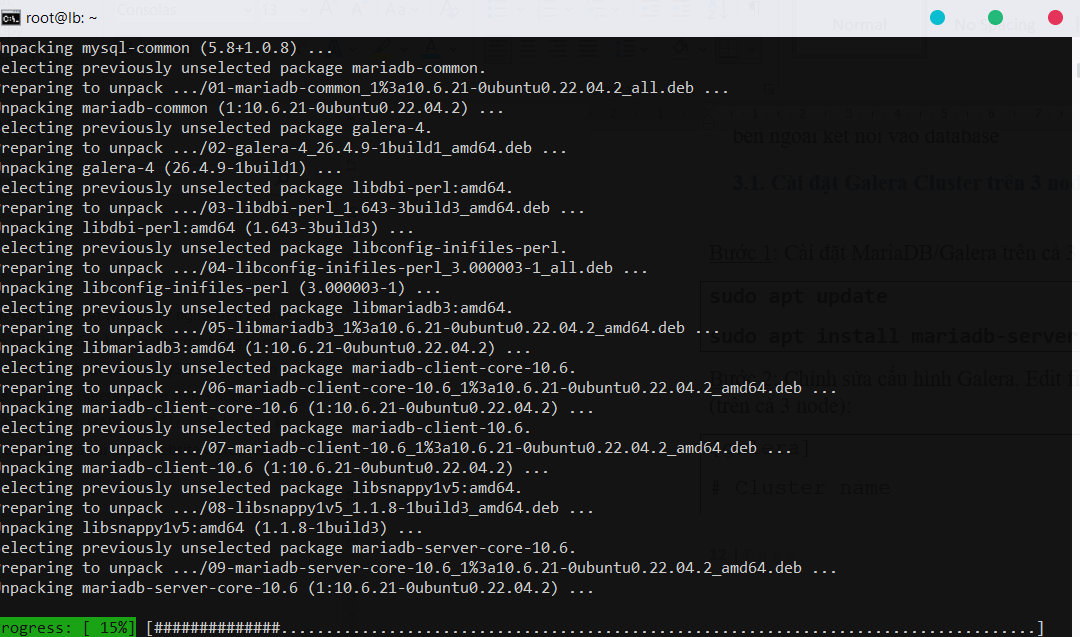
## Triển khai cluster database MySQL Galera trên 3 server web01, web02, và Database Server. Sau đó sử dụng Load Balancer ở câu 2 để cấu hình và điều phối traffic cho bên ngoài kết nối vào database

#### **3.1. Cài đặt Galera Cluster trên 3 node**

Bước 1: Cài đặt MariaDB/Galera trên cả 3 node

**sudo apt update**

**sudo apt install mariadb-server galera-4 -y**



Bước 2: Chỉnh sửa cấu hình Galera.

Edit tên cả 3 node:

sudo nano /etc/mysql/mariadb.conf.d/60-galera.cnf

sudo nano /etc/mysql/my.cnf

[galera]

# Cluster name

wsrep\_cluster\_name="mycluster"

# Cluster members

wsrep\_cluster\_address="gcomm://192.168.115.2,192.168.115.3,192.168.115.4"

# Node IP (set riêng cho mỗi node)

wsrep\_node\_address="192.168.115.2"

wsrep\_node\_name="nodeWeb01"

**wsrep\_provider=/usr/lib/galera/libgalera\_smm.so**

# MySQL settings

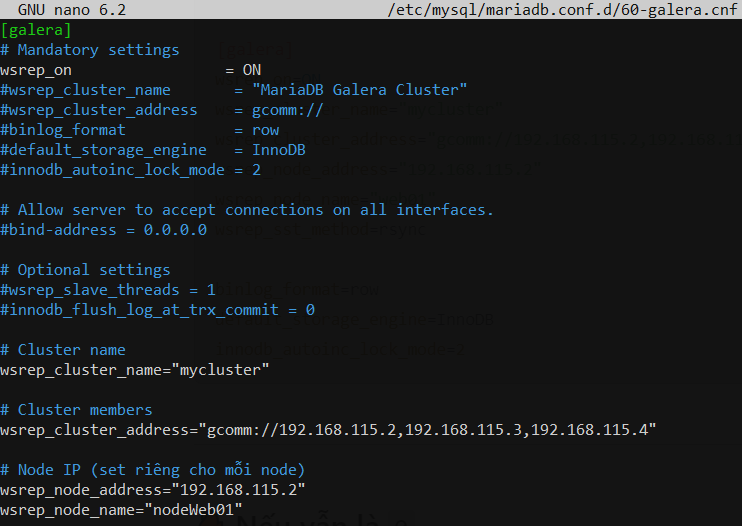
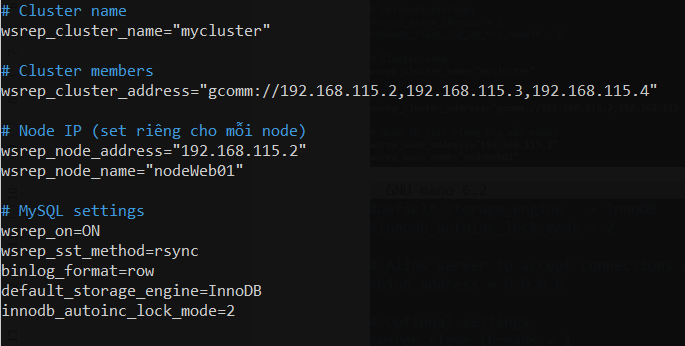
wsrep\_on=ON

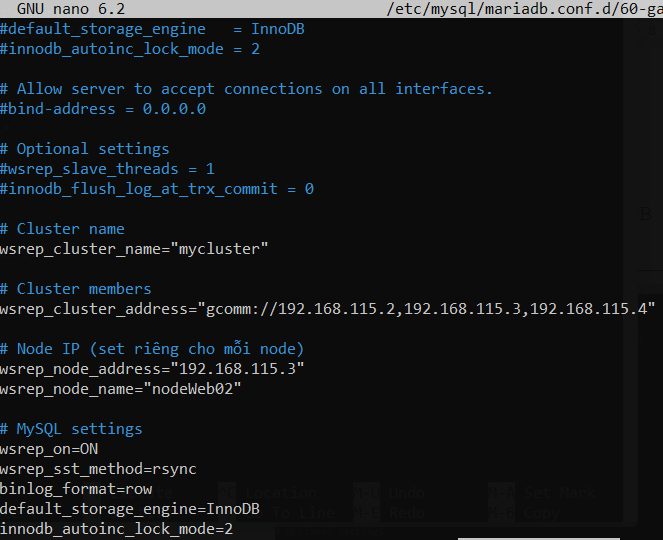
wsrep\_sst\_method=rsync

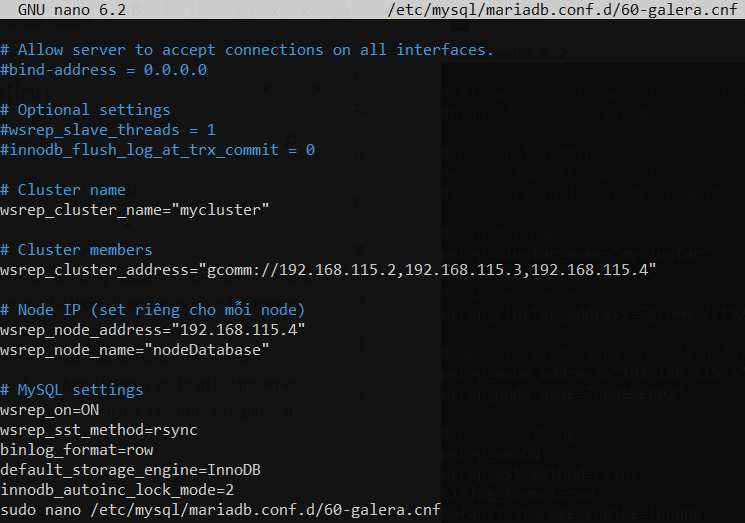
binlog\_format=row

default\_storage\_engine=InnoDB

innodb\_autoinc\_lock\_mode=2





Bước 3: Tắt MySQL trên cả 3 node trước khi khởi động cluster

**sudo systemctl stop mysql**

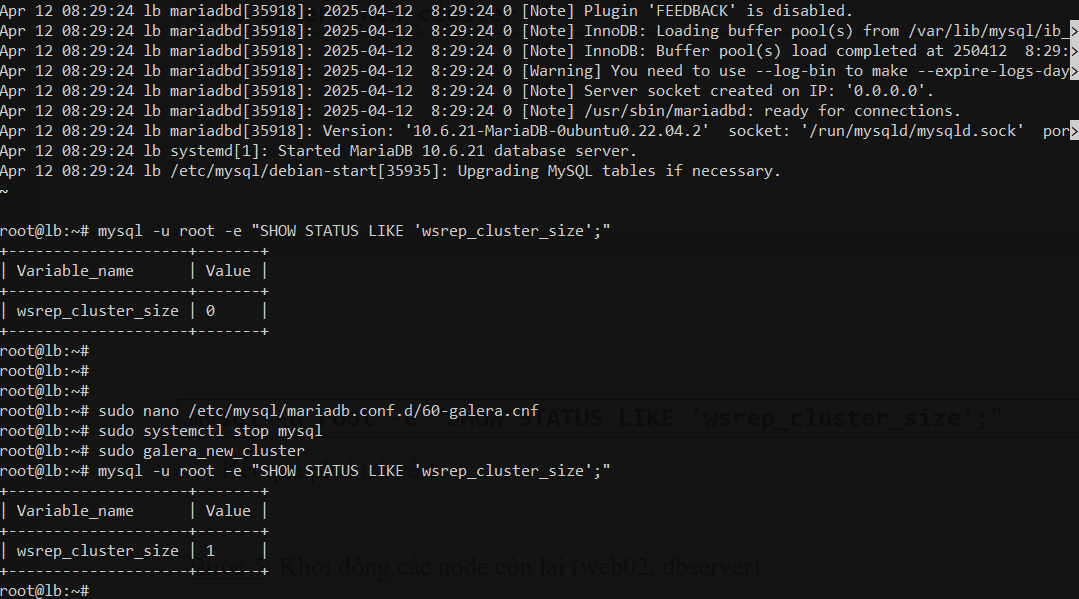
Bước 4: Khởi động cluster trên node đầu tiên (web01)

**sudo galera\_new\_cluster**

Kiểm tra:

**mysql -u root -e "SHOW STATUS LIKE 'wsrep\_cluster\_size';"**

→ Kết quả phải trả về 1



**Máy Web01**

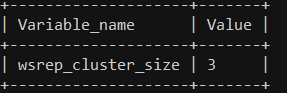
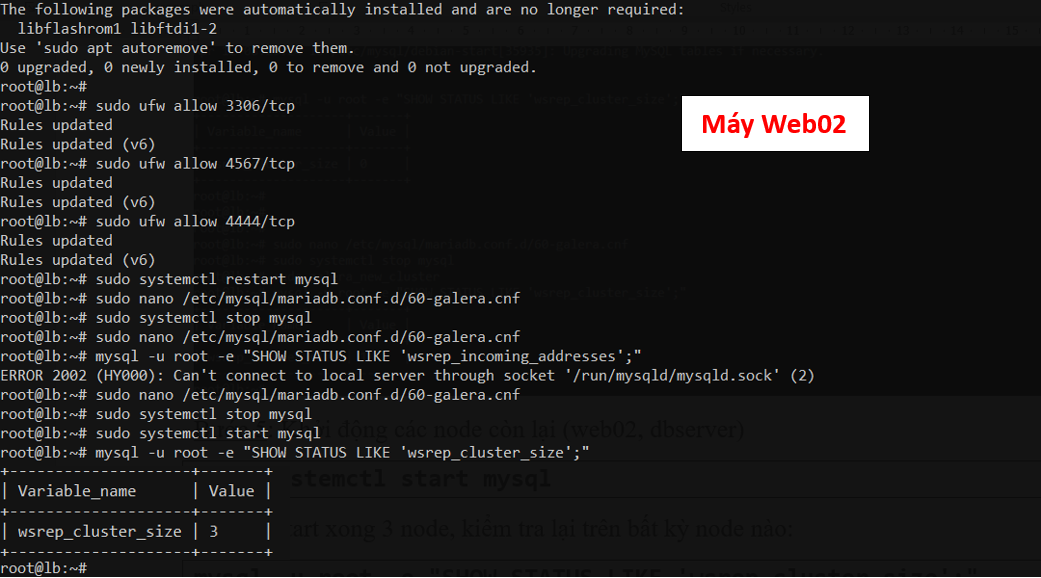
Bước 5: Khởi động các node còn lại (web02, dbserver)

**sudo systemctl start mysql**

Sau khi start xong 3 node, kiểm tra lại trên bất kỳ node nào:

**mysql -u root -e "SHOW STATUS LIKE 'wsrep\_cluster\_size';"**

→ Phải trả về 3 (số node trong cluster)



#### **3.2. Cấu hình HAProxy để Load Balancing kết nối MySQL**

Bước 1: Cài gói MySQL client trên LB (nếu cần test)

**sudo apt install mysql-client -y**

Bước 2: Thêm cấu hình vào /etc/haproxy/haproxy.cfg

listen galera-mysql

bind \*:3307

mode tcp

option mysql-check user haproxy\_check

balance roundrobin

server db1 192.168.115.2:3306 check

server db2 192.168.115.3:3306 check

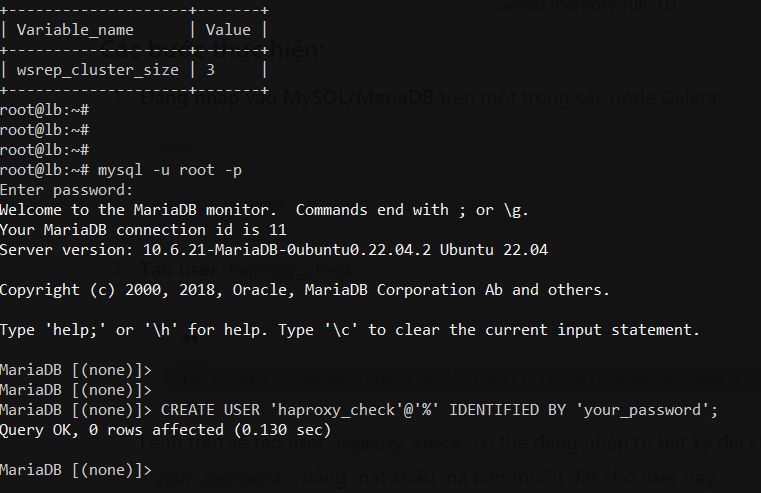
server db3 192.168.115.4:3306 check

Cổng 3307 là cổng mà LB lắng nghe — client bên ngoài sẽ kết nối vào LB:3307

Bước 3: Tạo user haproxy\_check trên một trong các node Galera

**CREATE USER 'haproxy\_check'@'%' IDENTIFIED BY 'your\_password';**

Lệnh trên sẽ tạo user haproxy\_check có thể đăng nhập từ bất kỳ địa chỉ IP nào (%), phần your\_password đặt password theo ý mình



**Máy Web01**

Cấp quyền cho user:

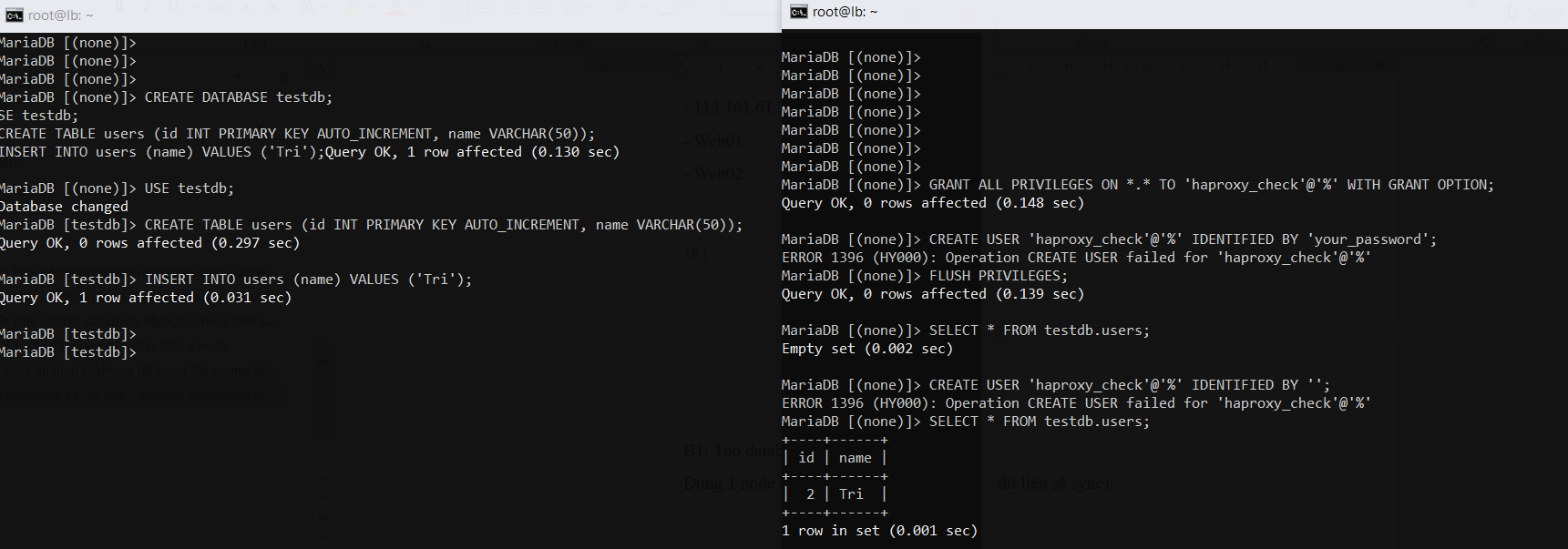
**GRANT ALL PRIVILEGES ON \*.\* TO 'haproxy\_check'@'%' WITH GRANT OPTION;**

Lệnh này sẽ cấp quyền toàn bộ cơ sở dữ liệu cho user haproxy\_check. Có thể tùy chỉnh quyền tùy theo nhu cầu (ví dụ: chỉ cấp quyền đọc trên các bảng cần thiết)

Tải lại quyền để thay đổi có hiệu lực:

**FLUSH PRIVILEGES;**

Sau khi thực hiện các lệnh trên trên một node Galera, việc đồng bộ hóa các thay đổi này sẽ xảy ra tự động qua Galera Cluster.



## Tạo database và cài đặt 1 website wordpress đơn giản với database vừa tạo. Phân quyền cho database chỉ cho 5 địa chỉ IP sau truy cập:

* Localhost
* 125.212.250.9
* 113.161.61.219
* Web01
* Web02

**B1: Tạo database + user cho WordPress**

Dùng 1 node bất kỳ trong Galera cluster (vì dữ liệu sẽ sync):

CREATE DATABASE wordpress\_db CHARACTER SET utf8mb4 COLLATE utf8mb4\_unicode\_ci;

CREATE USER 'wp\_user'@'localhost' IDENTIFIED BY 'StrongPass!123';

CREATE USER 'wp\_user'@'125.212.250.9' IDENTIFIED BY 'StrongPass!123';

CREATE USER 'wp\_user'@'113.161.61.219' IDENTIFIED BY 'StrongPass!123';

CREATE USER 'wp\_user'@'web01' IDENTIFIED BY 'StrongPass!123';

CREATE USER 'wp\_user'@'web02' IDENTIFIED BY 'StrongPass!123';

GRANT ALL PRIVILEGES ON wordpress\_db.\* TO 'wp\_user'@'localhost';

GRANT ALL PRIVILEGES ON wordpress\_db.\* TO 'wp\_user'@'125.212.250.9';

GRANT ALL PRIVILEGES ON wordpress\_db.\* TO 'wp\_user'@'113.161.61.219';

GRANT ALL PRIVILEGES ON wordpress\_db.\* TO 'wp\_user'@'web01';

GRANT ALL PRIVILEGES ON wordpress\_db.\* TO 'wp\_user'@'web02';

FLUSH PRIVILEGES;

Password: StrongPass!123

**B2: Cài đặt WordPress trên Web01 hoặc Web02**

Giả sử chọn Web01.

a) Cài PHP + dependencies

**sudo apt update**

**sudo apt install php php-mysql libapache2-mod-php -y**

b) Tải và giải nén WordPress

**cd /var/www/html**

**sudo rm index.html**

**sudo wget https://wordpress.org/latest.tar.gz**

**sudo tar -xzf latest.tar.gz**

**sudo mv wordpress/\* .**

**sudo rm -rf wordpress latest.tar.gz**

**sudo chown -R www-data:www-data .**

c) Tạo file cấu hình database

**cp wp-config-sample.php wp-config.php**

**nano wp-config.php**

Sửa các dòng sau:

define('DB\_NAME', 'wordpress\_db');

define('DB\_USER', 'wp\_user');

define('DB\_PASSWORD', 'StrongPass!123');

define('DB\_HOST', 'lb'); // hoặc IP của Load Balancer (hoặc 127.0.0.1 nếu test local)

(em làm không kịp)

