DEMO TRIỂN KHAI KEEPALIVE + HAPROXY CHO KIBANA

& FORWARD LOG CHO SERVER SỬ DỤNG DOCKER

Người thực hiện: Doãn Vĩnh Hoàng

**I. KEEPALVIE + HAPROXY CHO KIBANA**

*Mục lục:*

*1. Cấu hình của các dịch vụ trong mô hình*

*2. Kiểm tra trạng thái của các dịch vụ*

*3. Tiến hành kiểm tra*

*3.1 Tắt Kibana*

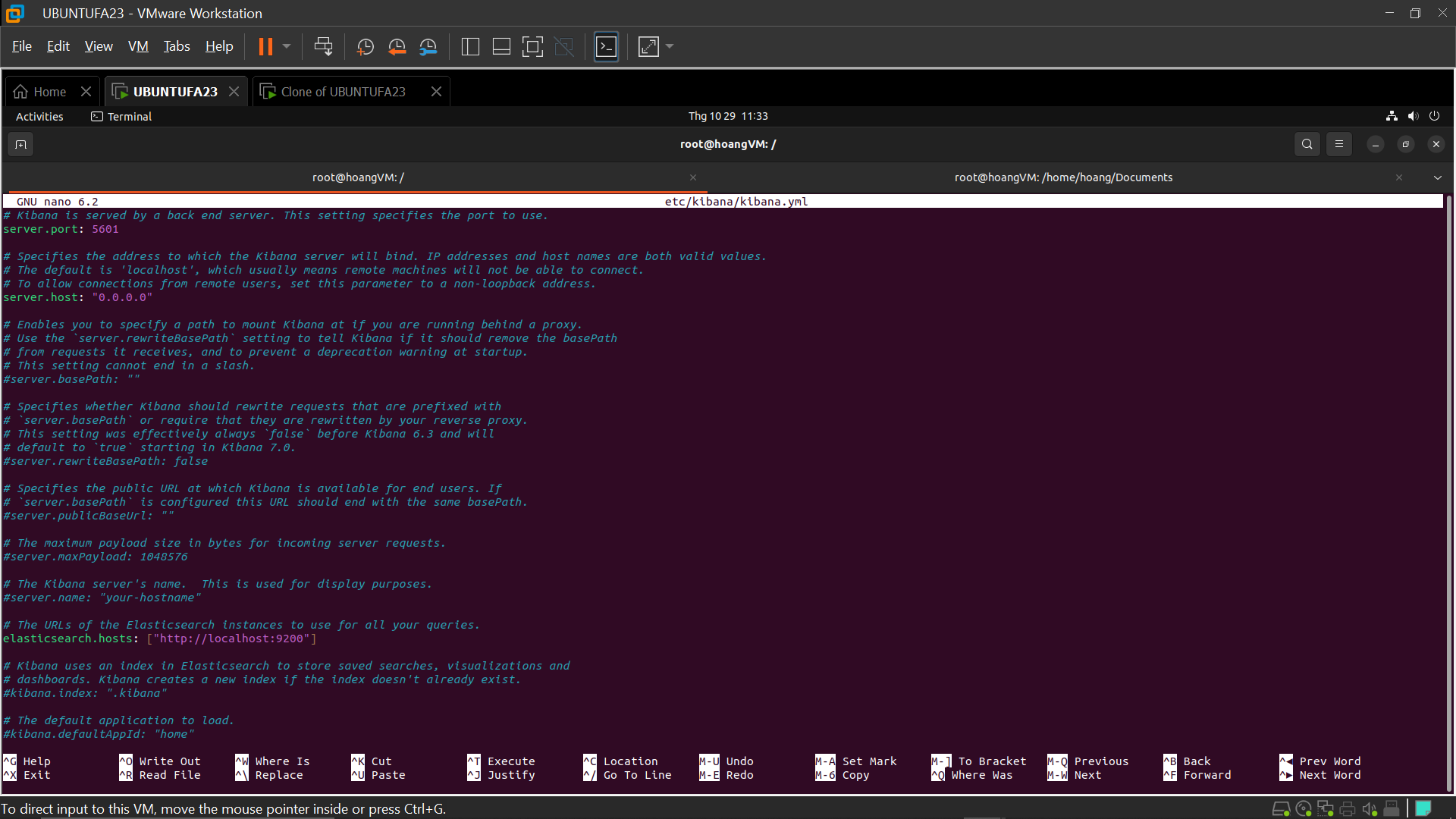
*3.2 Tắt server*

\*\*\*Trước khi bước vào demo: trong mô hình thật của Keepalived thì các log file sẽ được cập nhật liên tục theo thời gian thực và sẽ được thông qua docker và container cũng như các fluent-bit sẽ được nhận được thông tin giống nhau nhằm giữ được tính sẵn sàng cao. Tuy nhiên vì mục đích kiểm thử và demo nên hai service fluent-bit trên 2 máy ảo sẽ chỉ đọc file tĩnh và mỗi máy sẽ đọc một file khác nhau (để kiểm tra xem khi dịch vụ hoặc server down, keepalived và haproxy có hoạt động không).

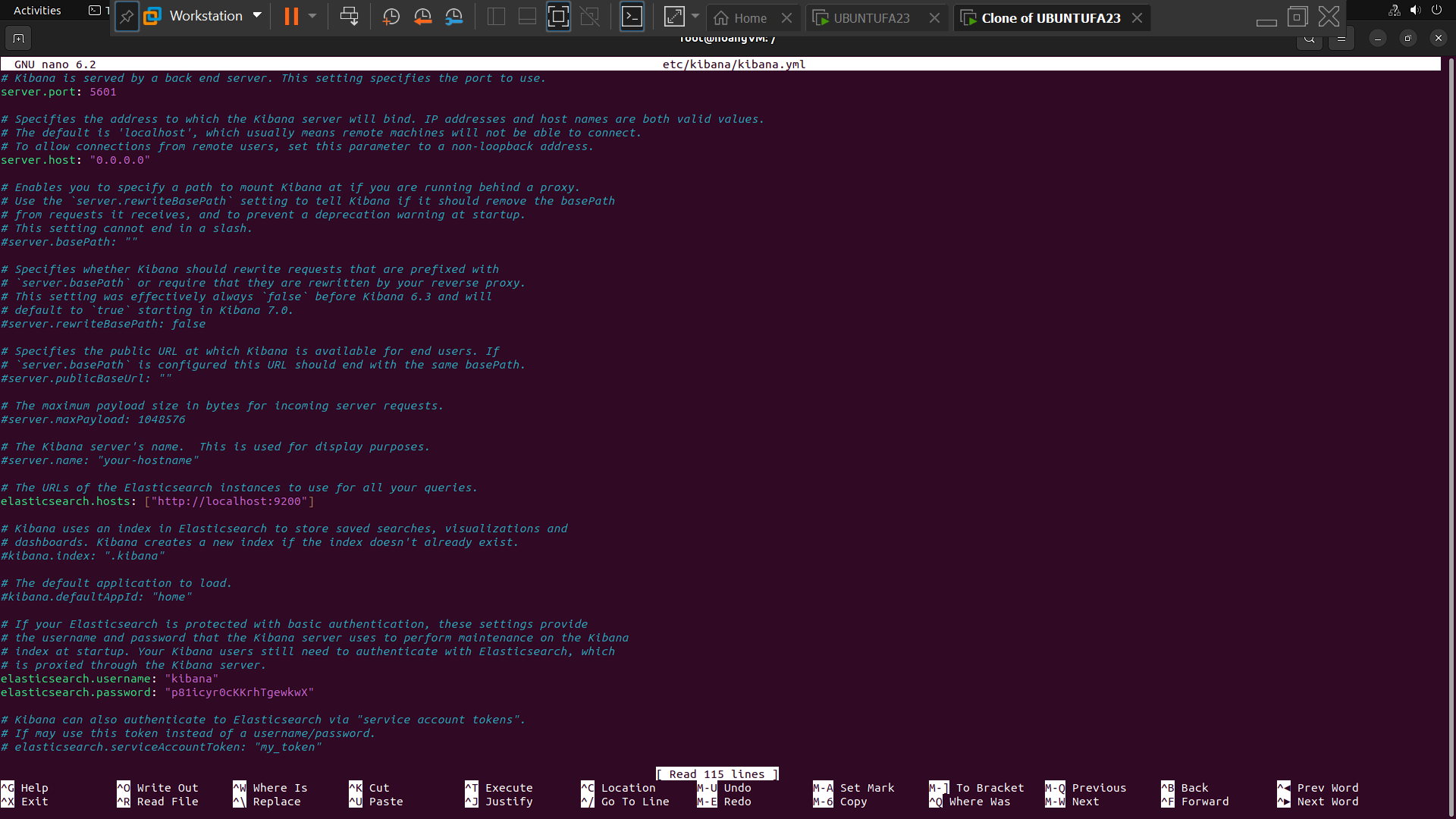
**1. Thông tin cấu hình từng dịch vụ trong mô hình**

**KIBANA:** chuyển server.host của cả hai sang 0.0.0.0 (tất cả địa chỉ)

+ VM1:

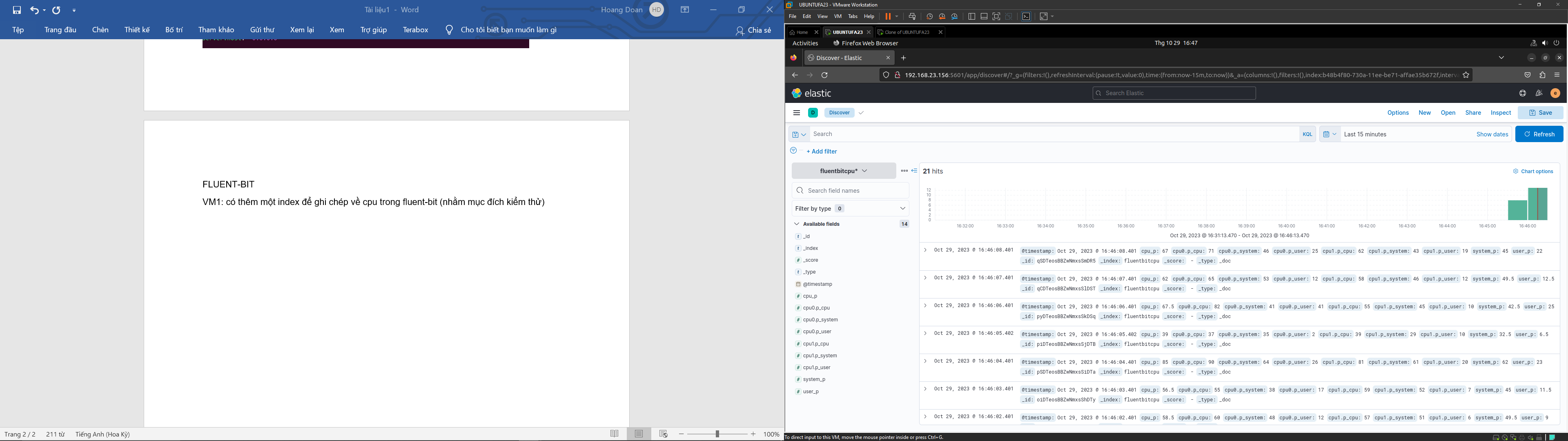


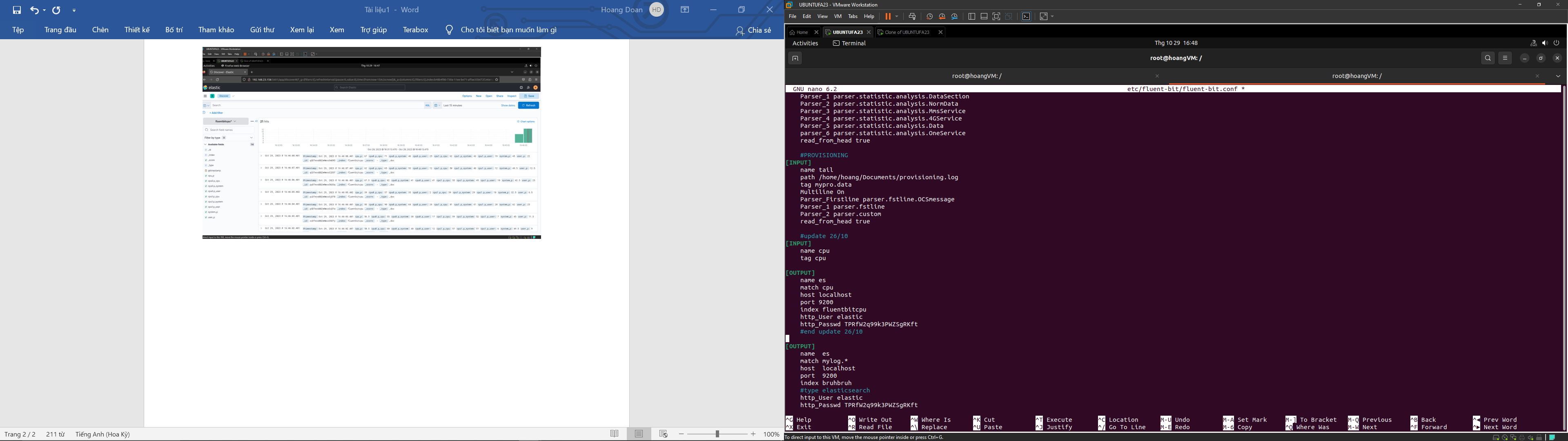
+ VM2:



**FLUENT-BIT:** ở đây sẽ xem trong phần fluent-bit.conf và kiểm tra trên kibana

+ VM1: có thêm một index để ghi chép về cpu trong fluent-bit (nhằm mục đích kiểm thử).





+ VM2: không có index trên.

**KEEPALIVED:**

|  |  |
| --- | --- |
| VM1 | VM2 |
|  |  |

- Trong phần cấu hình của cả VM1 và VM2, có thể thấy VM1 đã được set cho trạng thái MASTER và VM2 được set cho trạng thái BACKUP cùng với các thông số sau:

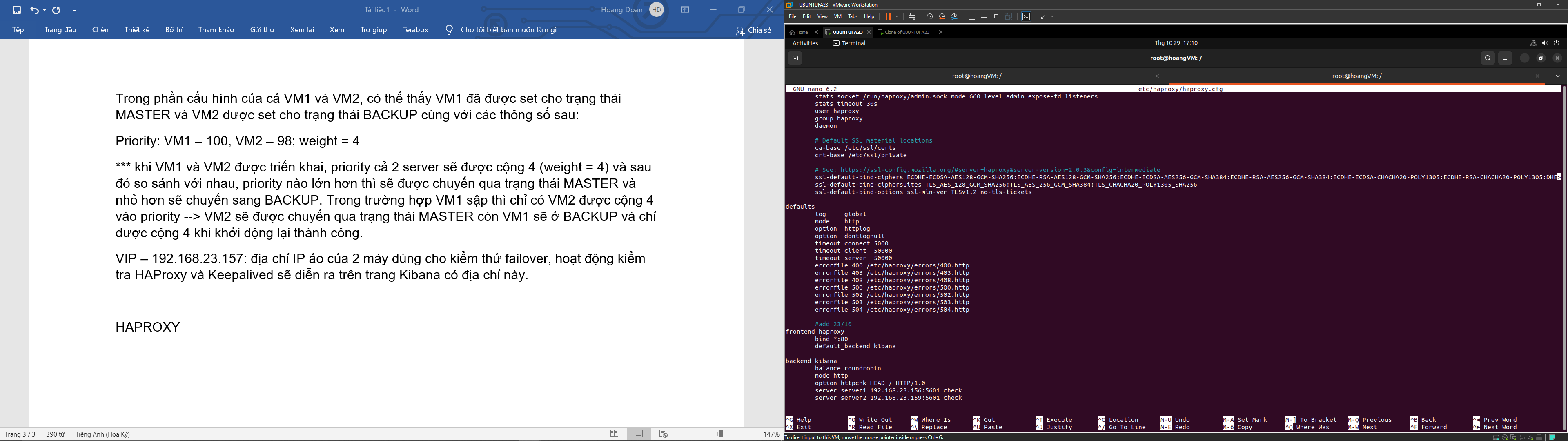
+ Priority: VM1 – 100, VM2 – 98; weight = 4

\*\*\* khi VM1 và VM2 được triển khai, priority cả 2 server sẽ được cộng 4 (weight = 4) và sau đó so sánh với nhau, priority nào lớn hơn thì sẽ được chuyển qua trạng thái MASTER và nhỏ hơn sẽ chuyển sang BACKUP. Trong trường hợp VM1 sập thì chỉ có VM2 được cộng 4 vào priority --> VM2 sẽ được chuyển qua trạng thái MASTER còn VM1 sẽ ở BACKUP và chỉ được cộng 4 khi khởi động lại thành công.

+ VIP – 192.168.23.157: địa chỉ IP ảo của 2 máy dùng cho kiểm thử failover, hoạt động kiểm tra HAProxy và Keepalived sẽ diễn ra trên trang Kibana có địa chỉ này.

**HAPROXY**

Cả hai cấu hình haproxy.conf trong hai máy ảo đều giống nhau:



+ server server1 192.168.23.156:5601 check : địa chỉ IPv4 của VM1.

+ server server1 192.168.23.156:5601 check : địa chỉ IPv4 của VM2.

+ check: HAProxy sẽ kiểm tra tình trạng của 2 server nhằm đảm bảo tính sẵn sàng cao.

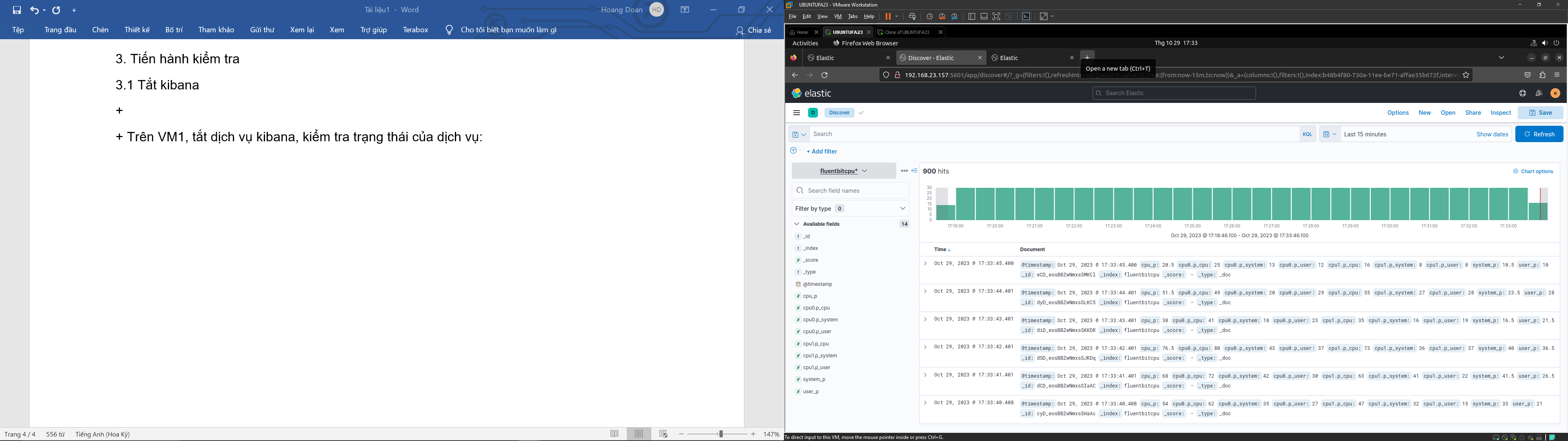
**2. Trạng thái của các dịch vụ khi thực hiện triển khai:**

|  |  |  |
| --- | --- | --- |
| Dịch vụ | Trạng thái | Thông tin thêm |
| kibana | VM1: active  VM2: active |  |
| fluent-bit | VM1: active  VM2: active | VM1 : nhận log cpu (index pattern : fluentbitcpu) |
| keepalived | VM1 : active  VM2 : active | VM1 : MASTER STATE  VM2 : BACKUP STATE |
| HAProxy | VM1 : active  VM2 : active | kibana/server1 is UP  kibana/server2 is UP |

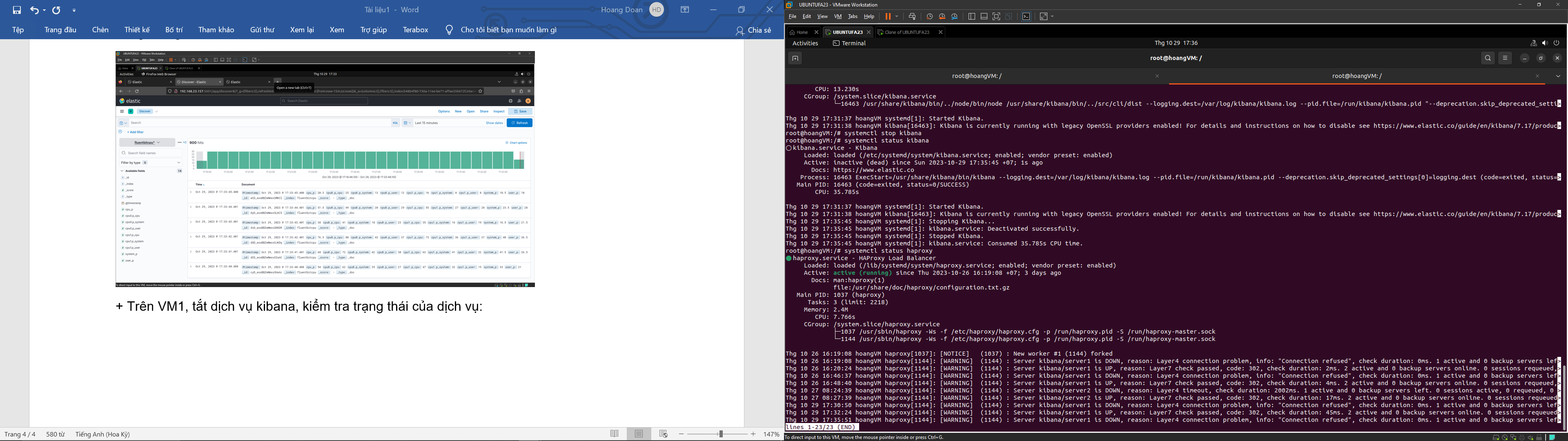
**3. Tiến hành kiểm tra**

**3.1 Tắt kibana**

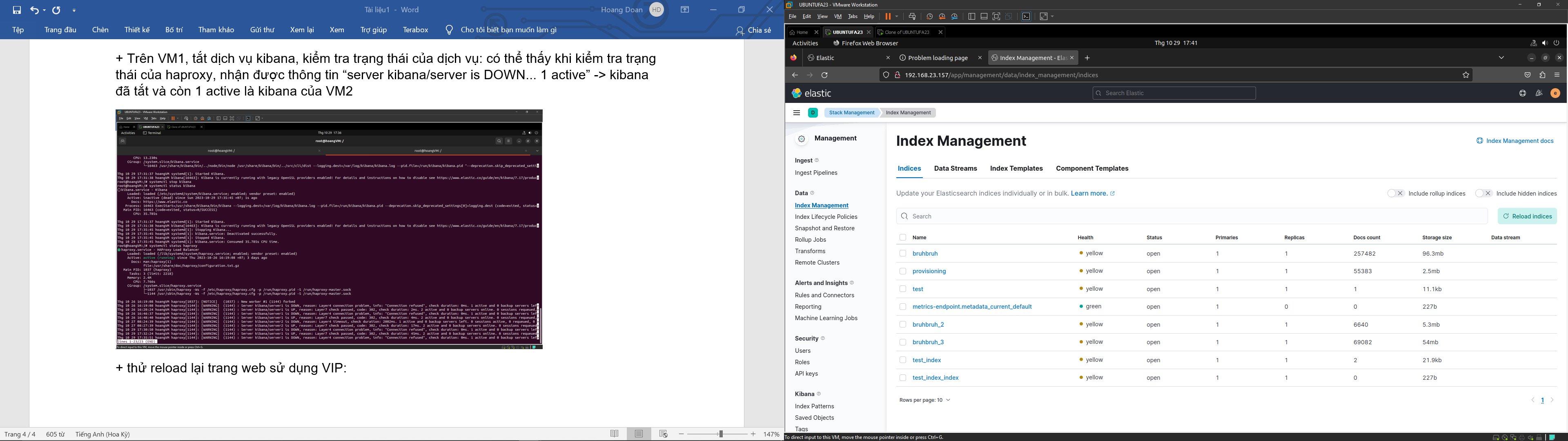
+ kiểm tra trang web bằng địa chỉ VIP, có log của index fluentbit CPU chứng tỏ web đang sử dụng thông tin của VM1



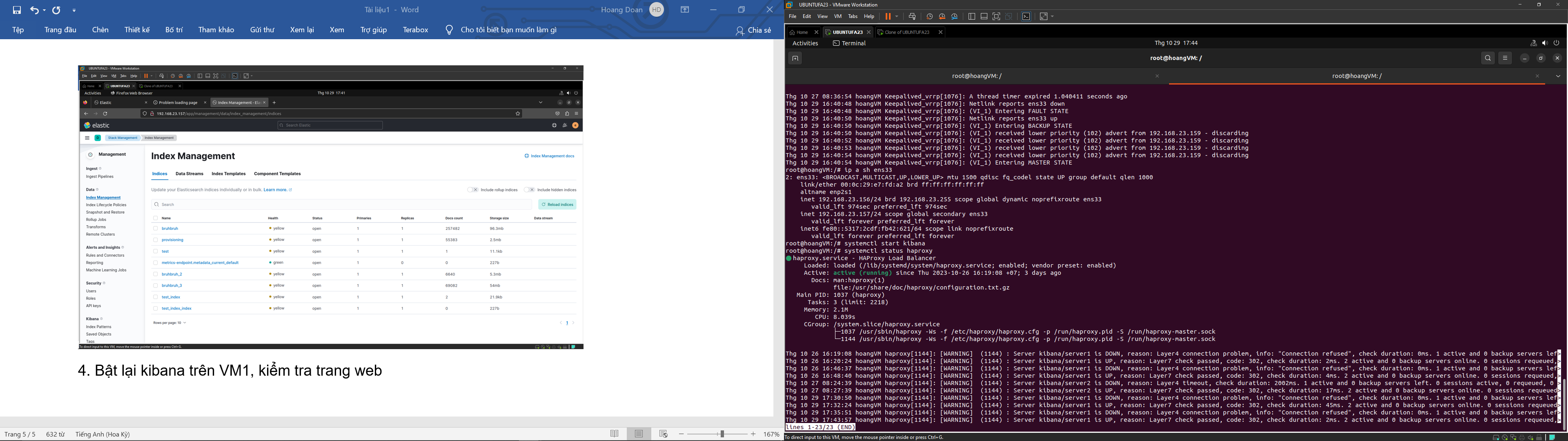
+ Trên VM1, tắt dịch vụ kibana, kiểm tra trạng thái của dịch vụ: có thể thấy khi kiểm tra trạng thái của haproxy, nhận được thông tin “server kibana/server is DOWN... 1 active” -> kibana đã tắt và còn 1 active là kibana của VM2



+ thử reload lại trang web sử dụng VIP: reload thành công vì đã chuyển qua kibana ở VM2 (không thấy fluentbitcpu trong phần Index Management)



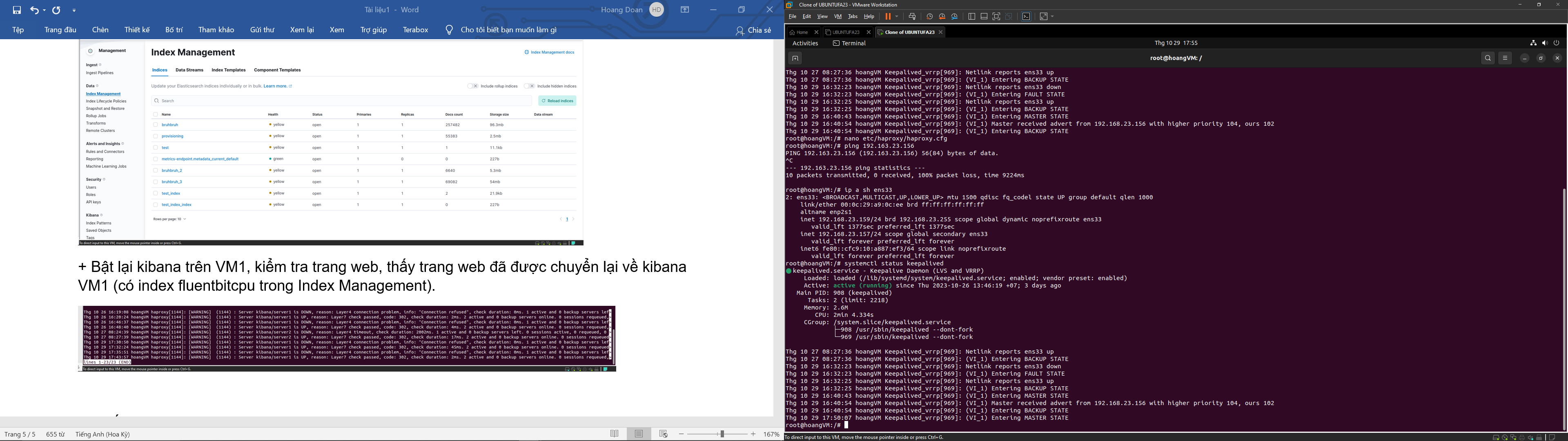
+ Bật lại kibana trên VM1, kiểm tra trang web, thấy trang web đã được chuyển lại về kibana VM1 (có index fluentbitcpu trong Index Management).



**3.2 Tắt server**

+ tắt VM1 (shutdown); Trên VM2, kiểm tra trạng thái server VM1 (192.168.23.156) và trạng thái của keepalived, thấy được VM2 đã được chuyển qua trạng thái MASTER

+ gõ lệnh ip a sh ens33 thì thấy Virtual IP đã được gán vào interface ens33 của VM2 -> VM2 đã được làm MASTER



+ kiểm tra trang web, truy cập thành công và cũng không thấy được index fluentbitcpu -> kibana và index được lấy từ VM2

+ khởi động lại VM1 sẽ thấy các trạng thái được chuyển về như ban đầu.

\*\*\* phần demo mô hình keepalived và haproxy trên EFK đã hoạt động thành công.