

# I PENDAHULUAN

## 1.1 Latar Belakang

Sekolah Vokasi IPB (SV-IPB) yang sebelumnya bernama Diploma IPB pertama kali dibuka oleh IPB pada tahun 1979 dan terletak di Jl. Kumbang No. 14 Bogor 16151. SV-IPB memiliki 17 Program Studi yang mana sistem pembelajarannya lebih memprioritaskan kepada praktikum dari pada teori. Kegiatan pembelajaran praktikum dilakukan di lab komputer, laboratorium atau di ruangan yang terbuka, dan untuk pembelajaran teori dapat dilakukan di ruangan kelas yang tersedia. SV-IPB juga melakukan kegiatan penelitian, pelatihan, dan pengabdian masyarakat yang berdasarkan tridarma perguruan tinggi dengan berbagai instansi baik pemerintah maupun swasta yang mana untuk mempersiapkan lulusan yang siap kerja.

Pada saat ini SV-IPB telah mengikuti perkembangan teknologi informasi dengan menerapkan sistem pembelajaran *online* atau biasa di sebut *E-learning*. *E-learning* adalah suatu sistem pendidikan yang mana memanfaatkan perkembangan teknologi informasi seperti internet dalam melaksanakan proses belajar mengajar. Pembelajaran *E-learning* di SV-IPB telah didukung oleh sebuah sistem perangkat lunak yaitu LMS (*Learning Management System*). Salah satu contoh, LMS yang sering digunakan dan bersifat *open source* adalah MOODLE (*Modular Object-Oriented Learning Environment*). LMS Moodle sangat membantu dalam pembelajaran *online* karena memiliki fasilitas yang dapat menunjang pembelajaran seperti pemberian materi, tugas, kuis, ujian, dan obrolan grup (Elyas 2018).

Mahasiswa SV-IPB yang mencapai 7000 orang pertahun dalam mengikuti proses pembelajaran atau ujian secara *online* akan mengunjungi situs SV-IPB yang telah terpasang LMS Moodle secara bersamaan. Hal ini menyebabkan tingginya permintaan dan bahkan dapat melebihi permintaan yang dapat ditampung oleh servernya. Oleh sebab itu, server akan mengalami kelebihan beban atau *overload* sehingga dapat mengakibatkan server tidak dapat lagi memproses permintaan yang ada. Server yang mengalami kegagalan sistem sehingga tidak bisa diakses lagi, biasanya disebut dengan *server down* (Nurhatta 2012; Prakash dan Deepalakshmi 2017; Hakim *et al.* 2019; Hanafiah 2021).

Permasalahan tersebut dapat diatasi dengan membuat sebuah *prototype* sistem menggunakan teknik *load balancing*. *Load balancing* adalah sistem yang dapat membagi beban permintaan dari *user* ke beberapa server agar tidak terjadi kelebihan beban pada salah satu server. Pembagian beban permintaan dari *user* akan merata ke semua server yang ada sehingga membuat kinerja server menjadi lebih mudah. Haproxy sebagai *load balancer* dapat dipergunakan untuk mengatur pembagian beban tersebut. Selain itu, haproxy juga termasuk *software* gratis yang dapat diandalkan dalam hal *high availability*, *load balancing*, dan proxy. Adapun salah satu algoritme yang saat ini sering digunakan untuk *load balancing* yaitu *least connection*. Algoritme ini bekerja dengan mendistribusikan permintaan dari *user* ke server yang memiliki jumlah koneksi aktif yang paling sedikit di antara server yang lain (Nasser dan Witono 2016; Arman *et al.* 2017; De La Cruz dan Goyzueta 2017; Prasetijo *et al.* 2017; Rahmatulloh dan MSN 2017; Syaquia dan Asmunin 2017; Singh dan Kaur 2018; Sholeh *et al.* 2019; Riskiono dan Pasha 2020).



## 1.2 Rumusan Masalah

Rumusan masalah dari Penerapan *Load balancing* pada Server menggunakan Algoritme *Least Connection* di Sekolah Vokasi IPB adalah:

- a. Bagaimana membangun *prototype* sistem *load balancing* agar *request* setiap *user* tetap dapat di teruskan ke server ketika salah satu server mengalami masalah (*down*)?
- b. Bagaimana mengirimkan *request* dari *user* ke server yang sedang memiliki jumlah koneksi aktif paling sedikit menggunakan teknik *load balancing*?

## 1.3 Tujuan

Tujuan dari Penerapan *Load balancing* pada Server menggunakan Algoritme *Least Connection* di Sekolah Vokasi IPB adalah:

- a. Membangun *prototype* sistem *load balancing* dengan haproxy sebagai *load balancer* agar *request* dari *user* tetap dapat diteruskan ketika terjadi masalah terhadap salah satu server (*down*).
- b. Mengimplementasikan *load balancing* menggunakan algoritme *least connection* agar *request* dari *user* dapat dikirimkan ke server yang memiliki jumlah koneksi aktif paling sedikit.

## 1.4 Manfaat

Manfaat dari Penerapan *Load balancing* pada Server menggunakan Algoritme *Least Connection* di Sekolah Vokasi IPB adalah:

- a. Memindahkan *request* dari *user* ke server yang masih aktif ketika server lain mengalami masalah (*down*).
- b. Mengurangkan beban dari satu server ke server lainnya secara merata.

## 1.5 Ruang Lingkup

Ruang lingkup dari Penerapan *Load balancing* pada Server menggunakan Algoritme *Least Connection* di Vokasi IPB adalah:

- a. Menggunakan Virtual Box untuk membangun *prototype* sistem *load balancing*.
- b. Menggunakan HAProxy sebagai *load balancer*.
- c. *Load balancing* menggunakan algoritme *least connection*.
- d. Protokol pengalamatan yang digunakan yaitu IP versi 4.
- e. Menggunakan Apache Benchmark untuk melakukan simulasi pengiriman *request* dari *user* ke server.
- f. Penerapan *load balancing* hanya pada protokol HTTP.
- g. Pengujian *Load balancing* dilakukan pada web LMS Moodle.

# II METODE

## 2.1 Lokasi dan Waktu PKL

Pelaksanaan Praktik Kerja Lapangan dilaksanakan di Sekolah Vokasi Institut Pertanian Bogor, Jl. Kumbang No. 14 Bogor 16151. Kegiatan Praktik Kerja