

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Load balancing merupakan suatu teknik untuk mendistribusikan beban *traffic* pada dua atau lebih koneksi secara seimbang, agar *traffic* dapat berjalan optimal dan menghindari *overload* pada koneksi. Tujuan mekanisme tersebut agar memaksimalkan *throughput*, memperkecil waktu tanggap dan menghindari *overload* pada suatu koneksi (Sirajuddin, 2012). *Load balancing* memiliki beberapa metode dalam implementasinya, seperti menggunakan algoritma *round robin* yang mampu membagi beban *traffic* dengan *server* yang berbeda. Peneliti ingin menggunakan metode tersebut berdasarkan sumber daya yang dimiliki oleh *server* yang dipilih pada saat pengguna melakukan permintaan, karena dengan mengambil sumber daya yaitu *response time* dan *throughput* yang paling ringan, kemungkinan meningkatkan kinerja dan meringankan beban *server*.

Dalam perkembangan paradigma jaringan, ditemukanlah paradigma baru bahwa penggunaan *software* untuk mengatur *traffic* dalam suatu jaringan yaitu paradigma *Software Defined Network* (SDN). SDN adalah sebuah teknologi jaringan dengan paradigma pemisah antara *control plane* dan *data plane* pada perangkat jaringan. *Control plane* berfungsi mengatur logika pada perangkat, sedangkan *data plane* berfungsi untuk meneruskan paket yang masuk ke suatu port menuju port tujuan. (Nugroho, 2017).

Implementasi algoritma *round robin* pada jaringan SDN memerlukan sebuah *controller*, dan ada beberapa jenis *controller* SDN saat ini, antara lain: Ryu, Pox, ONOS, OpenDayLight dan Floodlight. Setiap *controller* memiliki keunggulannya masing-masing dan dasar pemrograman yang berbeda. Perbedaan tersebut juga yang membuat masing-masing *controller* tersebut menghasilkan kinerja yang berbeda diukur dari *cpu usage*, *response time*, dan *throughput*.

Salah satu penelitian yang membahas implementasi *load balancing* pada jaringan SDN adalah penelitian yang dilakukan oleh Hafizhul Karim yang berjudul “Implementasi Load Balancing Web Server dengan Algoritme Weighted Least Connection pada Software Defined Network”. Pada penelitian tersebut menggunakan *server heterogen* (tidak sama) dan menggunakan parameter CPU Usage, Connection Rate, dan Throughput. Penentuan nilai beban (*weight*) ditentukan oleh *administrator*. *Administrator* dapat

mengkonfigurasi *server* mana yang dapat menampung koneksi yang banyak dan sedikit. Nilai beban secara *default* bernilai 1, tidak boleh bernilai 0. Pada *weighted least connection* nilai beban disesuaikan dengan spesifikasi *server*. Arsitektur jaringan yang digunakan yaitu SDN karena memberikan manajemen jaringan yang dinamis dalam melakukan konfigurasi jaringan terpusat pada *controller* (Karim, 2019).

Berdasarkan penjelasan diatas, algoritma tersebut tidak memperhatikan kepadatan *traffic* yang ada pada jaringan. Semakin tinggi tingkat kepadatan *traffic* jaringan maka akan semakin tinggi resiko terjadinya kegagalan komunikasi data tersebut menyebabkan paket yang dikirim mengalami keterlambatan dikarenakan adanya kenaikan *traffic* mendekati *throughput*. Oleh karena itu, penelitian ini bertujuan untuk mendistribusikan beban permintaan pengguna dengan pemilihan jalur berdasarkan *traffic* terendah menggunakan algoritma *round robin* pada *software defined network*. Pengimplementasian tersebut menggunakan *controller* SDN yaitu *Pox controller*. Parameter yang diujikan yaitu *CPU usage*, *throughput*, dan *response time*. Diharapkan penelitian ini mampu memberikan alternatif penentuan *load balancing* pada *software defined network*.

1.2 Perumusan Masalah

Berdasarkan dari latar belakang tersebut, maka dapat disimpulkan rumusan masalah sebagai berikut:

- a. Bagaimana kinerja *load balancing* dengan menggunakan algoritma *round robin* pada *POX controller* berdasarkan parameter *response time*?
- b. Bagaimana kinerja *load balancing* dengan menggunakan algoritma *round robin* pada *POX controller* berdasarkan parameter *CPU usage*?
- c. Bagaimana kinerja *load balancing* dengan menggunakan algoritma *round robin* pada *POX controller* berdasarkan parameter *throughput*?
- d. Bagaimana hasil analisa kinerja *load balancing* menggunakan algoritma *round robin* terhadap *server* di jaringan SDN?

1.3 Batasan Masalah

Adapun batasan masalah pada penelitian ini adalah:

- a. Penulis hanya menggunakan POX *controller*.
- b. Hanya mengimplementasikan algoritma *round robin*.
- c. Hanya menggunakan 1 *switch*.
- d. Topologi yang digunakan penulis adalah topologi *star*.
- e. Simulasi topologi dibuat dengan menggunakan Mininet.
- f. Pembuatan topologi menggunakan MiniEdit.

1.4 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

- a. Untuk mengetahui kinerja *load balancing* dari penggunaan algoritma *round robin* pada POX *controller* berdasarkan parameter *response time*.
- b. Untuk mengetahui kinerja *load balancing* dari penggunaan algoritma *round robin* pada POX *controller* berdasarkan parameter *CPU usage*.
- c. Untuk mengetahui kinerja *load balancing* dari penggunaan algoritma *round robin* pada POX *controller* berdasarkan parameter *throughput*.
- d. Untuk mengetahui hasil analisa kinerja *load balancing* dari algoritma *round robin* pada jaringan SDN.

1.5 Manfaat Penelitian

Manfaat yang dapat diambil dari skripsi ini adalah:

- a. Dapat mengetahui hasil kinerja *load balancing* dari algoritma *round robin* yang menggunakan POX *controller* melalui parameter yaitu *throughput*, *CPU usage*, dan *response time*.
- b. Beban *server* menjadi lebih ringan, karena beban yang dikirimkan oleh pengguna diterima oleh *server* lainnya.