

BAB I

PENDAHULUAN

1.1 Latar Belakang

Ethernet merupakan sebuah protokol pada *layer Data-link* yang banyak digunakan. Ethernet pada awalnya dikembangkan pada tahun 1970, oleh para peneliti di *Xerox Palo Alto Research Center* (PARC). Pada 1978 *Digital Equipment Corporation and Intel Corporation* bergabung dengan Xerox untuk menetapkan standar Ethernet pada 10 Mbps, yang mana kemudian standar ini dibentuk untuk dasar standar IEEE 802.3. Ethernet saat ini dapat mengirimkan informasi atau data dengan *bitrate* dari 10 Mbps sampai beberapa Gbps, dan mampu untuk diimplementasikan dengan menggunakan kabel *coaxial*, *fiber optic* dan media *wireless* [1].

Jaringan Ethernet menggunakan perangkat jaringan yang berupa *switch* untuk meneruskan data ke alamat *Media Access Control* (MAC) tujuan. *Switch* bekerja pada lapis *data-link*, sehingga ia tidak dapat mendeteksi alamat IP dan tidak memiliki mekanisme perutingan. Hal tersebut dikarenakan *switch* tidak memiliki tabel *routing* seperti pada perangkat *router*. Pada jaringan *backbone* internet saat ini, Ethernet digunakan bersamaan dengan *Internet Protocol* (IP), sebuah protokol pada lapis *network* yang mana memberikan mekanisme ruting dan layanan *forwarding*, sehingga paket dapat dikirimkan dari sumber ke tujuan yang mana dapat melalui jaringan yang berbeda. Namun, Ethernet hanya dapat meneruskan *frame* dengan mengenali alamat fisik atau MAC *address* tujuan. Maka diperlukan sebuah protokol *Address Resolution Protocol* (ARP) untuk mengartikan alamat IP ke alamat MAC, sehingga *frame* dapat diteruskan ke *hop* selanjutnya berdasarkan table *routing* [2]. Jika sebuah jaringan terdiri dari banyak *switch*, maka mekanisme *forwarding* nya menggunakan *Spanning*

Tree Protocol (STP) untuk meneruskan frame. Pada dasarnya, STP menggunakan satu jalur dalam mekanisme meneruskan frame. Jika hanya menggunakan satu jalur, maka akan banyak *link* di jaringan yang tidak digunakan.

Perutean hanya dengan satu jalur pada jaringan Ethernet ini dinilai kurang dapat memaksimalkan penggunaan *bandwidth* yang tersedia dan memberikan *delay* jaringan yang tinggi. *Delay* yang tinggi ini terjadi karena adanya *congestion* di jaringan. Hal tersebut dikarenakan mekanisme perutean STP yang digunakan. Dalam mekanisme perutean STP, trafik hanya dilewatkan melalui satu jalur yang sama dari sumber sampai tujuan. Trafik merupakan data yang berisi informasi, yang akan dibawa dan dikirimkan melalui jaringan data [3]. Jika trafik yang ditawarkan cukup besar dan melebihi kapasitas yang tersedia di jalurnya, maka trafik tersebut tidak dapat dilewatkan lagi pada jalur tersebut dan juga dapat mengakibatkan kualitas layanannya menurun. Oleh karena itu, perlu adanya sebuah metode untuk menggunakan jalur lebih dari satu untuk melewatkan trafik tersebut.

Salah satu solusi untuk mengatasi permasalahan tersebut adalah dengan menggunakan mekanisme perutean *multipath routing*. Tidak seperti mekanisme rute STP sebelumnya, yang hanya menggunakan satu jalur untuk melewatkan trafik, *multipath routing* membagi trafik untuk dilewatkan di beberapa jalur yang tersedia, serta akan lebih efisien daripada penggunaan mekanisme rute tradisional dengan satu jalur [4]. Jika pada perutean STP hanya menggunakan satu jalur untuk melewatkan trafik, maka akan banyak jalur yang tidak terpakai di dalam jaringan, yang berarti penggunaan *bandwidth* yang tersedia di jaringan tersebut menjadi kurang maksimal. Hal tersebut menjadikan penggunaan sumber daya jaringan seperti *bandwidth*, akan kurang maksimal. Oleh karena itu, skema *multipath routing* akan lebih dapat memaksimalkan penggunaan *bandwidth* dan memberikan *delay* yang rendah dari pada skema *single path routing* karena peruteannya yang menggunakan beberapa jalur. Selain itu, jaringan akan dapat mengantisipasi kesalahan ketika terjadi *link failure*. Tujuan dari penggunaan mekanisme *multipath routing* ini yaitu untuk memaksimalkan penggunaan *bandwidth* serta memberikan *delay* jaringan yang

rendah daripada mekanisme perutean STP sebelumnya. Dalam hal ini, terdapat beberapa metode dalam mekanisme perutean *multipath routing* yang sudah ada dan biasa digunakan, diantaranya *Equal-Cost Multi-Path* (ECMP) yang mana memilih jalur berdasarkan *minimum cost* yang sama dari 2 jalur, namun memiliki kekurangan dimana algoritma ECMP tidak mempertimbangkan kebutuhan *bandwidth* pada aliran trafik yang ditawarkan dan keadaan trafik yang sedang berjalan di keseluruhan jaringan, sedangkan *Load-Balanced Multi-Path* (LBMP) yang mana memilih jalur berdasarkan jalur dengan ketersediaan *bandwidth* yang paling besar yang tersedia di jaringan tersebut yang akan dialirkan trafik baru, sehingga beban trafik jaringan akan lebih seimbang.

Dalam tugas akhir ini, dilakukan perancangan algoritma *multipath routing* dengan teknik *Load-balanced Multi-Path* (LBMP) dan *admission control*. Teknik tersebut dipilih atas dasar karena lebih dapat memaksimalkan penggunaan *bandwidth* yang tersedia dan dapat membagikan beban trafik di jaringan dengan baik, serta memberikan *delay* yang rendah pada jaringan Ethernet. Dalam perancangan algoritma LBMP ini, digunakan paradigma *Software Defined Networking* (SDN). SDN memiliki fungsi yaitu memisahkan *control plane* dengan *user plane* [5], yang sebelumnya terdapat pada satu perangkat. Dengan SDN ini, informasi mengenai topologi jaringan dan pemantauan keadaan jaringan dapat diperoleh dengan efisien oleh kontroler *OpenFlow*.

1.2 Tujuan

Tujuan dari tugas akhir ini adalah merancang dan menganalisa algoritma *multipath routing* yang berbasis paradigma *Software Defined Networking* yang dapat memaksimalkan penggunaan *bandwidth* dan memberikan *delay* yang rendah pada jaringan. Dalam penelitian ini perancangan menggunakan Ryu sebagai kontroler, *OpenFlow* sebagai protokol komunikasi antara kontroler dengan perangkat *switch*, Mininet sebagai *prototyping environment*, sehingga kode dapat diaplikasikan ke jaringan nyata dengan atau tanpa perubahan kode.

1.3 Rumusan Masalah

1. Bagaimana perancangan algoritma *multipath routing* berbasis *Software Defined Networking* mampu memaksimalkan penggunaan *bandwidth* pada jaringan Ethernet, serta memberikan *delay* yang rendah pada jaringan Ethernet ?
2. Apakah *multipath routing* berbasis *Software Defined Networking* yang dirancang dapat memaksimalkan penggunaan *bandwidth* pada jaringan Ethernet, serta memberikan *delay* yang rendah pada jaringan Ethernet ?

1.4 Batasan Masalah

1. *Multipath routing* yang digunakan akan dirancang pada jaringan Ethernet.
2. Teknik yang dipakai untuk mendukung algoritma *multipath routing* adalah *load balancing* dan *admission control*.
3. Kontroler *OpenFlow* menggunakan Ryu versi 3.17
4. Pengerjaan tugas akhir ini diuji pada jaringan maya menggunakan Mininet 2.2
5. Protocol *OpenFlow* yang digunakan adalah versi 1.3
6. Topology yang digunakan adalah topology Abilene.

1.5 Metode Penelitian

Metode yang digunakan pada penelitian ini adalah:

1. Studi Literatur
Melakukan pendalaman materi yang terkait.
2. Perancangan
Melakukan perancangan algoritma dan pengkodean
3. Simulasi
Melakukan simulasi dan optimasi sehingga sesuai dengan spesifikasi yang diharapkan.
4. Pengujian dan Analisis

Melakukan pengukuran terhadap parameter-parameter: *delay*, *throughput*, *packet loss*, *convergence time* dan memory usage sesuai skenario yang telah ditentukan.

5. Pembuatan Laporan

Segala proses sejak perancangan hingga pengukuran dibukukan.

1.6 Sistematika Penulisan

Sistematika penulisan laporan adalah sebagai berikut:

- Bab 1 PENDAHULUAN
Berisi latar belakang, rumusan masalah, batasan masalah, tujuan, metode penelitian, sistematika penulisan
- Bab 2 DASAR TEORI
Berisi penjelasan teori, alat dan perlengkapan yang digunakan dalam penelitian
- Bab 3 PERANCANGAN DAN SIMULASI
Berisi alur kerja, alur perancangan, dan alur simulasi yang dilakukan dalam penelitian
- Bab 4 PENGUJIAN DAN ANALISIS
Berisi langkah pengujian yang dilakukan, hasil pengujian, dan analisis dari hasil pengujian yang didapat.
- Bab 5 KESIMPULAN DAN SARAN
Berisi kesimpulan dari tugas akhir ini. Berisi saran untuk penelitian selanjutnya.