BAB I

PENDAHULUAN

1.1 Latar Belakang

Umumnya pada perangkat *Router* atau *Switch*, *data plane* dengan *control plane* masih tergabung menjadi satu, dengan adanya paradigma baru dalam jaringan komputer yang memisahkan antara *data plane* dengan *control plane* yang disebut *Software Defined Network* (SDN) ini memungkinkan perangkat *Router* atau *Switch* untuk memprogram sesuai dengan yang diinginkan secara terpusat. Pengguna *Software Defined Network* (SDN) dapat membangun sendiri sebuah atau gabungan aplikasi yang akan dijalankan pada platform *controller* (*control plane*). Platform *controller* menyediakan *Application Programming Interface* (API) sehingga memudahkan dalam mengimplementasikan fitur dan layanan dalam jaringan komputer. Pada *Software Defined Network* ini, *controller* terpusat mengkonfigurasi *table forwarding* (*flow-table*) *Switch* atau *Router* yang bertugas untuk meneruskan aliran paket komunikasi.

Penelitian yang mengangkat permasalahan tentang *Software Defined Network* (SDN) sudah banyak dilakukan. Salah satunya adalah *Implementasi Software Defined Network Berbasis Opendaylight Menggunakan Switch Simulator Mininet* yang ditulis oleh Dani Ekaputra (2016). Pada penelitian tersebut penulis membuat simulasi jaringan *Software Defined Network* menggunakan *controller* yang bernama *Opendaylight*. *Controller* *Opendaylight* ini menggunakan bahasa pemrograman *Java* adalah bahasa pemrograman yang multi platform dan multi device. Aplikasi dengan berbasis *Java* ini dikompilasikan ke dalam *p-code* dan bisa dijalankan dengan *Java Virtual Machine*. Kelebihan dari *Java* ini adalah multi platform. *Java* dapat dijalankan dalam beberapa platform komputer dan sistem operasi yang berbeda. Selain itu *Java* memiliki *library* yang lengkap. *Library* disini adalah sebuah kumpulan dari program yang disertakan dalam *Java*. Kekurangan dari *Java* ini sendiri adalah penggunaan memori yang besar.

Dalam penelitian ini, penulis membuat simulasi jaringan *Software Defined Network* (SDN) dengan menggunakan *controller* yang berbeda dari penelitian sebelumnya. Adapun *controller* lain yang dapat digunakan dalam membangun jaringan *Software Defined Network* ini adalah Ryu Controller. Ryu Controller adalah kerangka kerja perangkat lunak jaringan dengan API yang terdefinisi dengan baik yang memudahkan pengembang untuk membuat aplikasi manajemen dan kontrol jaringan baru. Ryu Controller ini menggunakan bahasa pemrograman *python*. *Python* adalah bahasa pemrograman tingkat tinggi (*high-level programming language*), berjalan dengan sistem interpreted, dan bisa dipakai untuk berbagai jenis tujuan (*general-purpose*). *Python* disebut dengan bahasa pemrograman tingkat tinggi karena perintah atau kode program yang dipakai sudah mirip dengan bahasa manusia (bahasa Inggris). Kelebihan dari *python* ini adalah mudah dipelajari dan juga kode program yang dipakai lebih singkat. Kekurangan dari *python* ini sendiri adalah *python* tidak secepat atau efisien seperti bahasa pemrograman C.

Dalam membuat jaringan dibutuhkan topologi. Untuk membuat topologi pada jaringan *Software Defined Network* (SDN) ini dibutuhkan sebuah emulator untuk merancang sebuah sistem atau topologi jaringan. Salah satu emulator yang digunakan yaitu Mininet. Di dalam mininet terdapat topologi bawaan atau yang biasa disebut topologi *default*. Topologi yang digunakan penulis dalam penelitian ini adalah topologi *linear*, topologi *single* dan topologi *tree* yang merupakan topologi *default* dari Mininet.

Berdasarkan latar belakang dari permasalahan diatas, maka penulis membuat Proyek Akhir ini dengan judul *Implementasi Software Defined Network pada Ryu Controller menggunakan Switch Emulator Mininet*.

1.2 Rumusan Masalah

Adapun rumusan masalah pada Proyek Akhir ini adalah sebagai berikut :

1. Bagaimana cara implementasi *Software Defined Network* (SDN) pada Ryu Controller menggunakan *Switch Emulator Mininet* dengan pengujian pada 3 topologi?
2. Bagaimana performansi jaringan dari 3 topologi yang telah dibuat?

1.3 Batasan Masalah

Terdapat beberapa batasan masalah yang digunakan pada penelitian Proyek Akhir ini di antaranya:

1. Membangun jaringan *Software Defined Network* dengan menggunakan 3 topologi *default* (bawaan) dari Mininet yaitu topologi *linear*, topologi *tree*, dan topologi *single*.
2. Controller yang digunakan adalah Ryu Controller.
3. Menggunakan *OS Linux (Ubuntu)* pada *PC host*.
4. Servis jaringan yang akan diujikan pada penelitian ini adalah dengan tes konektivitas (*ping*) dan dengan mengukur parameter *throughput* dan *delay* dari 3 topologi *default*.

1.4 Tujuan Penelitian

Tujuan dari penelitian Proyek Akhir ini adalah sebagai berikut :

1. Untuk mengimplementasikan *Software Defined Network* (SDN) pada Ryu Controller menggunakan *Switch Emulator Mininet* dengan pengujian pada 3 topologi.
2. Untuk membandingkan performansi dari 3 topologi yang telah dibuat.

1.5 Manfaat Penelitian

Manfaat yang diharapkan dari penelitian ini adalah :

1. Dengan simulasi yang dibuat, diharapkan dapat mengimplementasikan *Software Defined Network* (SDN) pada Ryu Controller menggunakan *Switch Emulator Mininet* dengan pengujian pada 3 topologi.
2. Dengan simulasi ini, diharapkan pembaca dapat mengerti konsep dan cara kerja dari *Software Defined Network* (SDN).
3. Dengan simulasi ini, diharapkan arsitektur *Software Defined Network* (SDN) dapat menggantikan arsitektur jaringan tradisional yang masih digunakan di perusahaan-perusahaan, universitas, sekolah, dan tempat lainnya.

1.6 Metodologi Penelitian

Metode penelitian yang digunakan dalam penulisan Proyek Akhir ini adalah sebagai berikut :

1. Study Literature

Metode ini dilaksanakan dengan melakukan studi kepustakaan melalui membaca buku-buku, skripsi, dan jurnal yang dapat mendukung penulisan Proyek Akhir yang relevan mengenai Software Defined Networking (SDN), Ryu Controller, *flow table*, dan Mininet.

2. Diskusi

Metode ini dilaksanakan dengan melakukan diskusi dengan dosen pembimbing Proyek Akhir.

3. Perancangan dan Analisa Sistem

Pada tahap ini dilakukan perancangan *flow table* di Ryu Controller, perancangan topologi di Mininet dan mengkonfigurasi perangkat-perangkat virtual yang dibutuhkan ditopologi tersebut agar dapat berkomunikasi satu sama lain, serta menganalisa kebutuhan sistem yang akan dibuat.

4. Pengujian

Pada tahap ini dilakukan pengujian simulasi ini untuk mengetahui jaringan Software Defined Network (SDN) dapat berjalan atau berkomunikasi dengan baik pada setiap perangkat virtual nya.

1.7 Sistematika Penulisan

Dalam penulisan Proyek Akhir ini pembahasannya dibagi menjadi 5 (lima) bab yang memberikan uraian secara rinci agar lebih mudah untuk dipahami. Adapun sistematika penulisannya sebagai berikut:

BAB I PENDAHULUAN

Bab ini membahas mengenai latar belakang, rumusan masalah, batasan masalah, tujuan penelitian, manfaat penelitian, dan metodologi penelitian.

BAB II LANDASAN TEORI

Bab ini membahas teori-teori yang menjadi dasar penelitian Proyek Akhir serta penelitian terdahulu yang relevan dengan bidang yang diteliti termasuk prosedur, subjek, dan temuannya.

BAB III PERANCANGAN DAN ANALISA

Bab ini membahas perancangan serta tahap-tahap perancangan obyek penelitian serta alat dan bahan analisa, prosedur pelaksanaan analisa, dan tahapan Analisa.

BAB IV HASIL DAN PEMBAHASAN

Bab ini menjelaskan metode perancangan, perhitungan proses dan cara mengimplementasikan perancangan. Bab ini juga berisi mengenai hasil dan pembahasan berupa analisa hasil rancangan .

BAB V PENUTUP

Bab ini berisi mengenai kesimpulan – kesimpulan dari proses perancangan dan analisa hasil rancangan serta saran yang diperlukan berdasarkan keterbatasan-keterbatasan yang ditemukan selama pembuatan Proyek Akhir.