BAB I

PENDAHULUAN

1.1 Latar Belakang Masalah

Arsitektur jaringan tradisional memiliki keterbatasan dalam memenuhi permintaan aplikasi heterogen pada kebutuhan komunikasi yang terus meningkat. Pembatasan tersebut terjadi karena penerapan kebijakan spesifik waktu selama pembuatan perangkat yang membatasi fleksibilitas konfigurasi. Keputusan konfigurasi jaringan tradisional membutuhkan pemrosesan pada setiap perangkat seperti *router* atau *switch* (Y. Zhang dkk., 2018). Sehingga, menyulitkan dalam bidang flexibilities, kemampuan program, dan virtualisasi pada jaringan tradisional yang kompleks dan tertutup (Priyadarsini & Bera, 2021). Oleh karena itu, jaringan *Software Defined Networks* (SDN) semakin meningkatnya paradigmanya untuk dapat memenuhi tuntutan jaringan masa depan sekaligus mengatasi tantangan infrastruktur jaringan tradisional dalam komunikasi.

Arsitektur SDN memberikan kemudahan dalam mengkonfigurasikan jaringan serta menawarkan banyak manfaat termasuk manajemen, dinamika dan efektivitas biaya. Penambahan konfigurasi kebijakan fungsional baru dapat dilakukan pengontrol dengan bantuan API. Arsitektur API terdiri dari *northbound* dan *southbound* API. *Northbound* merupakan API untuk komunikasi antara *application layer* dengan *controller*, sedangkan *southbound* merupakan API untuk komunikasi antara *controller* dengan *infrastructure layer*. Aspek utama arsitektur jaringan SDN mencakup pemisahan *control plane* dan *data plane* yang dapat diprogram secara langsung. Hal ini memberikan kemampuan *interface* yang dapat diprogramkan, otomatisasi, dan *control* jaringan yang memungkinkan operator untuk membangun konfigurasi jaringan dengan fleksibel dalam memperluas fungsi sumber daya jaringan secara mudah dan cepat (Ateya dkk., 2019).

Pada saat ini, SDN menjadi elemen kunci *evolusioner* dalam transformasi pada dunia jaringan. Adopsi SDN terus meningkat di berbagai sektor industri untuk mencapai infrastruktur jaringan yang adaptif. Penerapan SDN dalam meningkatkan kinerja skalabilitas jaringan telah banyak digunakan di lingkungan industri selama

beberapa tahun terakhir, baik untuk koneksi kabel maupun nirkabel. SDN menjadi paradigma jaringan *evolusioner* yang telah diadopsi oleh penyedia jaringan dan *cloud* (Khorsandroo dkk., 2021). SDN memungkinkan untuk beradaptasi terhadap pola lalu lintas kompleks yang memberikan efisiensi dalam mengoptimalkan kinerja jaringan. Akan tetapi, keamanan jaringan SDN menjadi tantangan dalam perkembangan terkini. Peningkatan keamanan dapat membantu mengatasi kekhawatiran kerahasiaan data dalam integrasinya di lingkungan SDN. Kondisi keamanan SDN terkini menjadi cerminan industri untuk terus beradaptasi dalam meningkatkan keamanan yang terus berkembang pada arsitektur jaringan SDN.

Arsitektur jaringan SDN yang terpusat, terbuka dan dapat diprogramkan menimbulkan kerentanan keamanan dalam manajemen jaringan. Pemisahan lapisan *control* dari lapisan *forwarding* menyebabkan ketergantungan pada *control* yang terpusat (Y. Zhang dkk., 2018). Jaringan SDN yang tersentralisasi dan terdistribusi menyebabkan ketergantungan pada *controller* terpusat menjadi kelemahan potensial SDN terhadap serangan DDoS. Hal ini menjadi celah bagi penyerang untuk dapat meluncurkan serangan DDoS (*Distributed Denial of Service*) antara komunikasi *controller* dan *switch* yang membuat jaringan *down* (Y. Zhang dkk., 2018). *Controller* sebagai komponen utama pada SDN berpotensi mengalami target serangan para penyerang. Semua lalu lintas yang tidak diketahui harus dikirimkan ke *controller* untuk diselidiki. Sehingga, lalu lintas berbahaya dapat menyebabkan terjadinya serangan DDoS pada jaringan SDN. Serangan DDoS ditujukan untuk membuat sumber daya sistem tidak tersedia bagi pengguna yang sah. Sistem *controller* yang terkena serangan DDoS menyebabkan kelebihan beban lalu lintas sehingga mengakibatkan *server down. Controller* yang diserang oleh penyerang mempermudah dalam modifikasi *flow table*, mengakibatkan perubahan data, pemblokiran akses dan pelanggaran data yang menyebabkan munculnya aplikasi berbahaya pada bidang aplikasi (X. Li dkk., 2019).

Arsitektur *single controller* pada jaringan SDN tidak dapat memenuhi kebutuhan pengendalian dan pengelolaan dalam skala besar karena kemampuan yang terbatas. *Single controller* menyebabkan kemacetan pada satu titik *controller* yang mengakibatkan penurunan fleksibilitas dapat mempengaruhi kemampuan

manajemen lalu lintas (Saleh dkk., 2022). Serangan DDoS pada *control plane* menggunakan *single controller* menjadi hambatan SDN ketika *openflow switch* berkecepatan tinggi bekerja pada *data plane*. Pada *openflow switch* kebijakan dalam *forwarding packet* disimpan dalam *flow table*. *Packet in openflow switch* mencocokkan *header packet* pada *flow table* dengan entri aliran dan mengirimkan yang tidak cocok ke *controller* dengan pesan *packet in* (Gebremeskel dkk., 2023).

Keterbatasan arsitektur *single controller* dalam menangani *packet in* dengan jumlah besar yang terindikasi serangan DDoS mengakibatkan beban kerja yang tinggi dalam satu titik kegagalan pada c*ontroller*. Arsitektur *single controller* berpotensi *buffer overload* pada *openflow switch*. Hal ini terjadi karena *switch* menunggu respons dari *controller*. Respons arsitektur *single controller* yang berurutan terhadap latensi aplikasi sensitif pada *controller* menyebabkan kelebihan beban yang dapat meningkatkan latensi responsivitas. *High availability* yang responsivitas pada jaringan SDN dalam situasi *downtime* saat terindikasi serangan DDoS tidak dapat diterima dalam pengendalian *single controller* karena tidak mampu memenuhi kebutuhan jaringan yang luas (Y. Zhang dkk., 2018).

Skema *main-backup controller* menggunakan arsitektur *multi-controller* berbasis *load balancer* dalam penelitian ini dilakukan untuk meningkatkan pertahanan terhadap satu titik kegagalan dan skalabilitas dalam mengurangi latensi respons yang tinggi pada *controller*. Implementasi *load balancer* berperan pada jaringan SDN dalam situasi kelebihan beban pada *controller* dengan mendistribusian beban *traffic* pada *backup controller*. Fitur *extract* serangan yang digunakan dalam penelitian ini untuk mendeteksi kelebihan beban yang terindikasi sebagai serangan pada *controller* berdasarkan lonjakan *source port*. Ketika SDN dalam kondisi kelebihan beban, *port* sumber akan berubah lebih tinggi melebihi nilai *threshold*. Beban yang tinggi menyebabkan *overload* pada *openflow switch* yang menyebabkan kemacetan. Sehingga perlunya implementasi *multi-controller* berbasis *load balancer* untuk pertahanan dalam skalabilitas responsivitas *controller*.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, maka peneliti merumuskan masalah sebagai berikut:

1. Bagaimana pengaruh arsitektur *multi controller* berbasis *load balancer* dalam responsivitas *controller*?
2. Bagaimana pengaruh arsitektur *multi controller* berbasis *load balancing* dalam menjaga *high availability* pada kualitas jaringan (QoS) SDN?
3. Bagaimana analisis kinerja responsivitas dan *high availability multi-controller* berbasis *load balancer* dibandingkan dengan *single controller* pada jaringan SDN?

1.3 Tujuan penelitian

Penelitian ini memiliki tujuan sebagai berikut:

1. Untuk mengukur pengaruh arsitektur *multi-controller* berbasis *load balancing* dalam responsivitas *controller*.
2. Untuk mengukur kualitas jaringan (QoS) SDN pada saat serangan DDoS.
3. Untuk menganalisis responsivitas dan *high availability controller* berbasis arsitektur *multi-controller* dan *load balancer* dibandingkan dengan *single controller* pada jaringan SDN.

1.4 Batasan Penelitian

Penelitian ini memiliki batasan untuk mendapatkan hasil penelitian yang sesuai dengan tujuan penelitian. Berikut susunan batasan pada penelitian ini:

1. Penelitian ini menggunakan jenis serangan DDoS IP *spoofing*.
2. Penelitian ini menggunakan ryu *controller* dengan maximal 3 *controller.*
3. Penelitian ini melakukan pengukuran parameter performansi *responsivitas controller* dan *high availability* yang mencakup *latency, jitter,* dan *throughput.*

1.5 Manfaat Penelitian

Adapun Manfaat penelitian sebagai berikut:

1. Manfaat Teoritis

Penelitian ini diharapkan mampu memberikan pengetahuan mengenai pembagian *traffic* jaringan secara merata di berbagai *controller*, dalam mempertahankan skalabilitas dan ketahanan terhadap serangan.

2. Manfaat Praktis

a. Bagi Penulis

Meningkatkan keterampilan teknis keamanan jaringan informasi, dan teknologi SDN.

b. Pengembang Ilmu

Diharapkan dapat menjadi referensi acuan pengembangan sistem pengamanan jaringan SDN dari serangan.

1.6 Struktur Organisasi Skripsi

Berikut merupakan struktur organisasi skripsi pada penelitian ini diantaranya:

A. Bab 1: Pendahuluan

   Bab ini berisi gambaran awal penelitian dengan struktur latar belakang, identifikasi masalah, rumusan masalah, tujuan penelitian, batasan masalah, dan struktur organisasi skripsi.

B. Bab 2: Kajian Pustaka

   Bab ini terdiri atas landasan penelitian yang mendeskripsikan teori-teori penelitian berkenaan dengan masalah yang diteliti.

C. Bab 3: Metode Penelitian

   Bab ini mengarahkan pembaca untuk mengetahui alur dan desain penelitian, serta teknik pengumpulan data.

D. Bab 4: Hasil dan Pembahasaan

   Bab ini terdiri atas uraian hasil dan pembahasan penelitian yang telah dilakukan berdasarkan hasil pengolahan dan analisis data.

E. Bab 5: Penutup

   Bab ini terdiri atas kesimpulan, saran dan rekomendasi dari hasil pengukuran yang diperoleh dari penelitian.