

## 1. Pendahuluan

### Latar Belakang

*Distributed Controller* merupakan *multiple-controller* yang digunakan untuk meningkatkan skalabilitas dalam jaringan. Skalabilitas merupakan salah satu yang harus diperhatikan dalam suatu sistem, jaringan dan proses yang menunjukkan kemampuannya untuk menangani jumlah peningkatan kinerja yang berpotensi besar. *Distributed Controller* dibagi menjadi dua arsitektur yaitu *flat architecture* dan *hierarchical architecture*. *Flat architecture* merupakan *controller* yang diposisikan secara horizontal sehingga *control plane* hanya terdiri dari satu *layer* dan masing-masing *controller* memiliki tanggung jawab yang sama pada waktu yang sama [1]. *Flat architecture* memberikan lebih banyak ketahanan terhadap kegagalan dalam jaringan dibandingkan dengan *hierarchical architecture* karena memiliki beberapa *layer* dengan tanggung jawab yang berbeda sehingga lebih rentan mengalami kegagalan terhadap jaringan [2]. *Flat architecture Distributed Controller* dibagi menjadi dua yaitu, *active-active* dan *active-passive*. *Distributed Controller (active-active)* dengan menggunakan *flat architecture* membuat semua *controller* aktif dan bekerja sama untuk mengelola suatu jaringan secara bersamaan serta dapat memperluas kapabilitas dari *control plane* itu sendiri.

Pada penelitian sebelumnya dilakukan penelitian *Distributed Controller (active-active)* yang menggunakan *synchronous message exchange*. Dibandingkan dengan metode *active-backup* penelitian tersebut menghasilkan presentase kinerja pada CPU lebih rendah mengakibatkan proses failover menjadi lebih cepat [3]. Secara umum *synchronous message* memiliki kekurangan yaitu membutuhkan waktu yang lebih banyak ketika menunggu respon dari *receiver* karena komunikasi yang bersifat *real-time* dan beban kerja pada setiap *controller* lebih besar. Berbeda dengan *asynchronous message*, beberapa pesan dapat dikirimkan secara langsung kemudian *receiver* akan mengeksekusi. Penelitian ini merupakan pengembangan dari penelitian sebelumnya yaitu *asynchronous message*. Proses yang akan dilakukan yaitu menggunakan metode *3 in 1* yang bekerja dengan cara mengirimkan tiga pesan kemudian akan direspon oleh *receiver* dengan satu *acknowledgment*. Dengan menggunakan metode tersebut dapat mengurangi beban kerja masing-masing *controller* dan proses pengiriman pesan lebih cepat.

### Topik dan Batasannya

Dalam penelitian sebelumnya menggunakan *synchronous message* sebagai *message exchange* pada *Distributed Controller (active-active)*. Proses ini dilakukan secara *real-time* yaitu dengan mengirim satu pesan kemudian akan segera dilakukan *reply* dan begitupun seterusnya. Hal ini mengakibatkan penurunan kinerja, beban kerja terhadap *controller* lebih besar serta membutuhkan waktu yang lebih banyak. Oleh karena itu, penelitian ini mengusulkan untuk mengembangkan mekanisme *message exchange* tersebut dengan *asynchronous message* menggunakan metode *3 in 1*. Metode ini dilakukan dengan cara mengirimkan tiga pesan kemudian di *reply* dengan satu *acknowledgment* begitupun seterusnya. Dengan menggunakan metode ini dapat mengurangi beban kerja pada setiap *controller* serta waktu yang dibutuhkan lebih cepat.

Pada penelitian ini melakukan pengujian menggunakan parameter CPU *Usage* untuk menganalisa kinerja beban kerja (*workload*) terhadap *controller* serta menggunakan parameter dengan mekanisme *failover time* untuk melihat waktu *failover* yang terjadi ketika melakukan *switch migration* terhadap *controller*. Kemudian parameter *throughput* untuk membandingkan kinerja pada kedua metode *message exchange*. *Controller* yang digunakan pada penelitian ini adalah *POX controller*. Kemudian, dalam melakukan simulasi pada penelitian ini menggunakan tiga *virtualbox* yang digunakan sebagai *controller 1*, *controller 2*, dan juga *mininet*.

### Tujuan

Tujuan dilakukannya penelitian ini adalah menganalisis kinerja *asynchronous message exchange* pada *distributed controller (active-active)* dengan menggunakan metode *3 in 1* terhadap beban kerja serta waktu dalam proses *message exchange* dan membandingkannya dengan komunikasi *synchronous message*. Parameter yang digunakan dalam penelitian ini meliputi nilai CPU *usage* dan *failover* time untuk menganalisa beban kerja terhadap *controller* serta waktu *failover* yang terjadi. Kemudian, parameter *throughput* untuk melakukan perbandingan kinerja metode *3 in 1* dan satu pesan satu *acknowledgment (synchronous message)*.

### Organisasi Tulisan

Bagian selanjutnya terdiri atas studi terkait, sistem yang di bangun, evaluasi, dan kesimpulan. Pada bagian studi terkait akan dijelaskan studi literatur dan teori yang menjadi dasar pengembangan dan pelaksanaan penelitian ini. Penjelasan mengenai rancangan dan implementasi sistem akan dijelaskan lebih lanjut pada bagian sistem yang dibangun. Dari implementasi tersebut akan dilaksanakan beberapa skenario untuk melakukan pengujian terhadap sistem. Hasil dan analisis berdasarkan uraian pada bagian evaluasi juga dituliskan pada bagian kesimpulan.