

BAB V

KESIMPULAN DAN SARAN

3.1. Kesimpulan

Setelah melakukan penelitian tentang perancangan dan implementasi *Network Functions Virtualization* berbasis *cloud computing* dengan OpenStack, terdapat beberapa kesimpulan yang berhubungan dengan tujuan penelitian. Berikut kesimpulan yang dapat penulis jabarkan.

1. Berhasil merancang dan mengimplementasikan OpenStack sebagai NFVI.
2. Berhasil membuat *template* VNF Descriptor dan melakukan pra-konfigurasi VNF pada OpenStack Tacker.
3. Berhasil mengimplementasikan layanan pada VNF OpenWRT sebagai vRouter dan vFW pada OpenStack dengan Tacker.
4. Berhasil membuat perangkat lunak pengelolaan NFV untuk mengelola sistem infrastruktur NFV berbasis *cloud computing* dengan OpenStack.
5. Melakukan analisa hasil eksperimen dengan kesimpulan bahwa kinerja VNF dengan parameter *throughput packet* dan *CPU Utilization* berbanding terbalik dengan lamanya waktu pengujian. Hal ini disebabkan oleh perbedaan jumlah *traffic* yang diterima oleh VNF, kondisi jaringan, dan kemampuan NIC pada *client* dan *server*.
6. Terdapat limitasi pada API OpenStack Tacker, yang mengakibatkan tidak semua fungsi OpenStack Tacker dapat diimplementasikan baik pada perangkat lunak maupun *platform* OpenStack itu sendiri, seperti API tidak dapat menerima pra-konfigurasi pada *template* VNF Descriptor yang dibuat.
7. Pra-konfigurasi tidak selalu berhasil dilakukan. Hal ini disebabkan oleh *image* OpenWRT yang tidak selalu dapat menerima *user data*, sehingga diperlukan konfigurasi manual.
8. Saat ini, OpenStack Tacker masih dalam tahap pengembangan, dengan siklus pembaruan yang cukup cepat. Sehingga setiap perubahan tersebut dapat mempengaruhi fungsi-fungsi dari setiap API OpenStack Tacker.

3.2. Saran

Dalam pelaksanaan penelitian, penulis menyadari bahwa masih banyak kekurangan yang dilakukan oleh penulis dalam penelitian ini. Oleh karena itu, penulis menyampaikan beberapa saran yang dapat dilakukan di kemudian hari. Penelitian selanjutnya dapat menghasilkan perancangan konfigurasi sistem yang lebih baik, dan sebuah perangkat lunak yang memiliki fitur yang lebih kompleks. Berikut beberapa saran yang dapat penulis anjurkan.

1. Perangkat lunak dapat dibangun menggunakan beberapa *framework* yang berbasis Python seperti Django REST *Framework*, agar setiap API OpenStack dapat terintegrasi dengan mudah dan pemrosesan yang lebih cepat.
2. Untuk keperluan produksi dan pengembangan lebih lanjut, sebaiknya menggunakan *distro* OpenStack lain seperti OPNFV yang menyediakan OpenStack Tacker pada awal instalasi.
3. Mengukur skalabilitas terhadap VNF, dengan parameter CPU *Utilization* yang dapat diterapkan pada OpenStack Tacker.
4. Diharapkan penelitian ini bisa terus berlanjut dan bisa dikembangkan di bidang manapun