

**UJI THROUGHPUT KONTROLER FLOODLIGHT DAN BEACON
MENGUNAKAN EMULATOR MININET**

SKRIPSI



disusun oleh

Wikansetya Luhur Pambudi

12.11.5831

**JURUSAN TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
AMIKOM YOGYAKARTA
YOGYAKARTA
2015**

**UJI THROUGHPUT KONTROLER FLOODLIGHT DAN BEACON
MENGUNAKAN EMULATOR MININET**

SKRIPSI

Untuk memenuhi sebagian persyaratan
mencapai derajat Sarjana S1
pada jurusan Teknik Informatika



disusun oleh

Wikansetya Luhur Pambudi

12.11.5831

**JURUSAN TEKNIK INFORMATIKA
SEKOLAH TINGGI MANAJEMEN INFORMATIKA DAN KOMPUTER
AMIKOM YOGYAKARTA
YOGYAKARTA
2015**

PERSETUJUAN

SKRIPSI

**UJI THROUGHPUT KONTROLER FLOODLIGHT DAN BEACON
MENGUNAKAN EMULATOR MININET**

Yang dipersiapkan dan disusun oleh

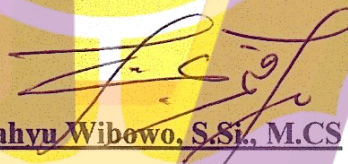
Wikansetya Luhur Pambudi

12.11.5831

Telah disetujui oleh Dosen Pembimbing Skripsi

Pada tanggal 12 Oktober 2015

Dosen Pembimbing,



Ferry Wahyu Wibowo, S.Si, M.CS

NIK. 190302235

PENGESAHAN

SKRIPSI

UJI THROUGHPUT KONTROLER FLOODLIGHT DAN BEACON MENGUNAKAN EMULATOR MININET

Yang disusun oleh

Wikansetya Luhur Pambudi

12.11.5831

Telah dipertahankan di depan Dewan Penguji

Pada tanggal 30 November 2015

Susunan Dewan Penguji

Nama Penguji

M. Rudyanto Arief, MT
NIK. 190302098

Melwin Syafrizal, S.Kom, M.Eng
NIK. 190302105

Ferry Wahyu Wibowo, S.Si., M.CS
NIK. 190302235

Tanda Tangan



Skripsi ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Sarjana Komputer
Tanggal 10 Desember 2015



KETUA STMIK AMIKOM YOGYAKARTA

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

PERNYATAAN

Saya yang bertandatangan dibawah ini menyatakan bahwa, skripsi ini merupakan karya saya sendiri (ASLI), dan isi dalam skripsi ini tidak terdapat karya yang pernah diajukan oleh orang lain untuk memperoleh gelar akademis di suatu institusi pendidikan tinggi manapun, dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis dan/atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Segala sesuatu yang terkait dengan naskah dan karya yang telah dibuat adalah menjadi tanggungjawab saya pribadi.

Yogyakarta, 2 Desember 2015



Wikansetya Luhur Pambudi
12.11.5831

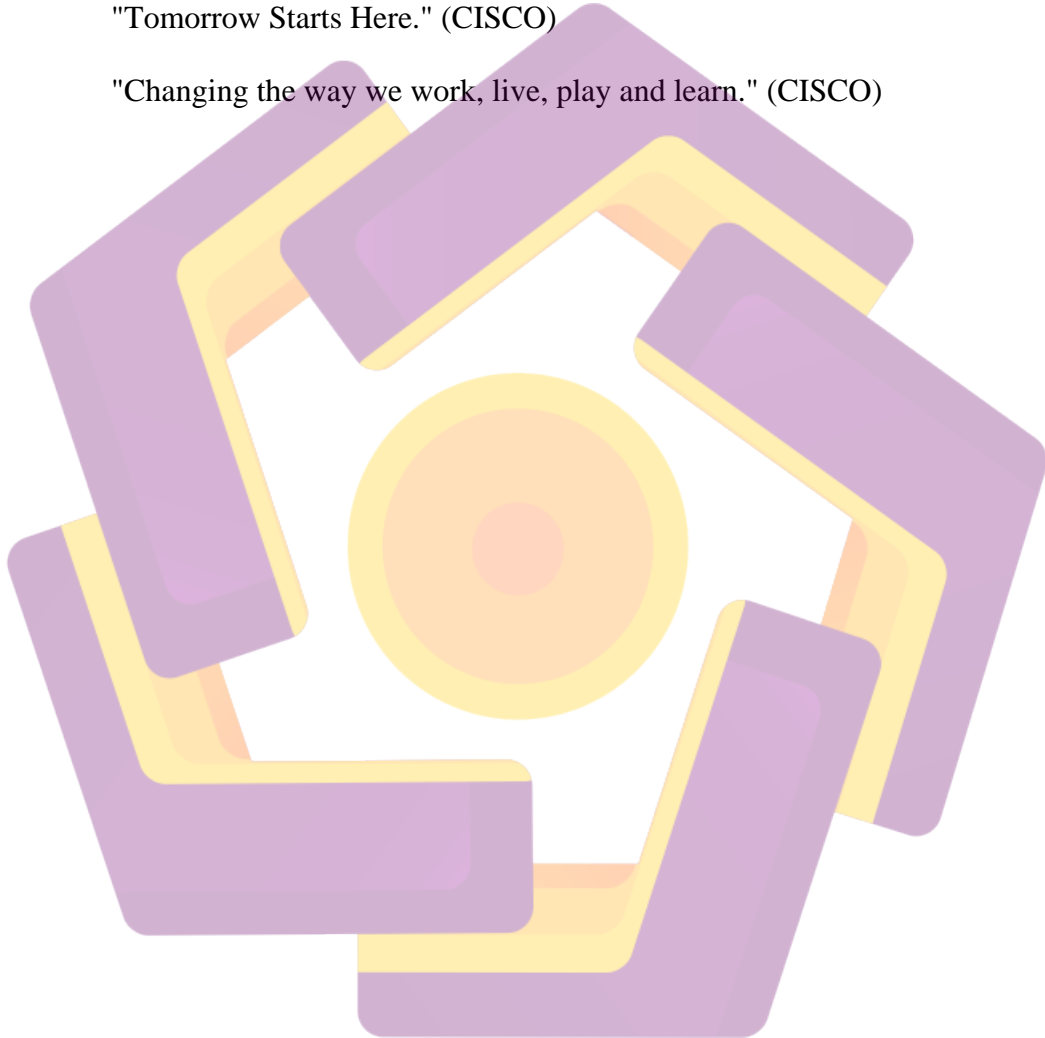
MOTTO

“Never give up, PUSH IT until your limit !”

“IT is easy and we make IT is easier.”

"Tomorrow Starts Here." (CISCO)

"Changing the way we work, live, play and learn." (CISCO)



PERSEMBAHAN

Kupersembahkan Skripsi Ini Teruntuk:

Allah SWT dan Rasulullah SAW

Sembah sujud serta syukur kepada Allah SWT yang telah menciptakanku, memberikan karunia, kenikmatan, dan perlindungan yang tak ternilai. Serta sholawat dan salam selalu terlimpahkan kepada Rasulullah SAW yang telah memberikan pengetahuan ajaran Tuhanku dan membawaku dari jurang kedengkian menuju kehidupan penuh keindahan.

Mama dan Papa tercinta

Sebagai tanda bakti, hormat, dan rasa terima kasih yang tidak terhingga kepada Mama dan Papa yang selalu mendoakanku, memberikan segala kasih sayangnya dengan penuh ketulusan dan yang telah berjuang dengan penuh keikhlasan hingga aku menyangang sarjana.

Adik-adikku tersayang

Sebagai tanda cinta kasihku kepada adik-adikku, walaupun sering bertengkar tapi hal itu ungkapan kasih sayang kakak kepada adiknya. Maaf karena belum bisa menjadi panutan yang baik untuk kalian.

Sahabat-sahabat grup PONY

Yang selalu memotivasi, canda, dan tawa saat bersama memberikan keringanan saat pengerjaan skripsi ini.

Teman-teman TI-02

Trimakasih atas dukungan dari teman-teman TI 02.

Almamaterku STMIK AMIKOM Yogyakarta

Tempat kuliahnya orang berdasi, mengajarkan menjadi seorang profesional harus diawali dengan merubah penampilan layak orang profesional.

KATA PENGANTAR

Puji syukur penulis panjatkan kepada Allah SWT yang telah melimpahkan rahmat, hidayah, serta karunia-Nya sehingga penulis dapat menyelesaikan skripsi ini tepat pada waktunya. Penyusunan skripsi ini bertujuan untuk memenuhi sebagai persyaratan guna memperoleh gelar Sarjana Teknik Informatika.

Penulis menyadari bahwa dalam penyusunan skripsi ini tidak lepas dari adanya kerjasama dan bantuan dari berbagai pihak. Oleh karena itu, pada kesempatan ini perkenankanlah penulis mengucapkan terima kasih kepada:

1. Bapak Prof. Dr. Mohammad Suyanto, M.M. selaku ketua STMIK AMIKOM Yogyakarta,
2. Bapak Ferry Wahyu Wibowo, S.Si., M.CS selaku dosen pembimbing,
3. Kedua orang tua penulis yang selalu memberikan kasih sayangnya,
4. Teman dan sahabat yang senantiasa mendukung dan memotivasi,
5. Serta semua pihak yang tidak bisa kami sebutkan satu persatu, yang telah memberikan dorongan serta bantuannya.

Penulis juga minta maaf atas segala kekurangan dan kesalahan dalam pembuatan skripsi ini. Tak luput dari itu kritik dan saran akan penulis tampung guna melengkapi kekurangan dan memperbaiki kesalahan.

Yogyakarta, 2 Desember 2015

Penulis

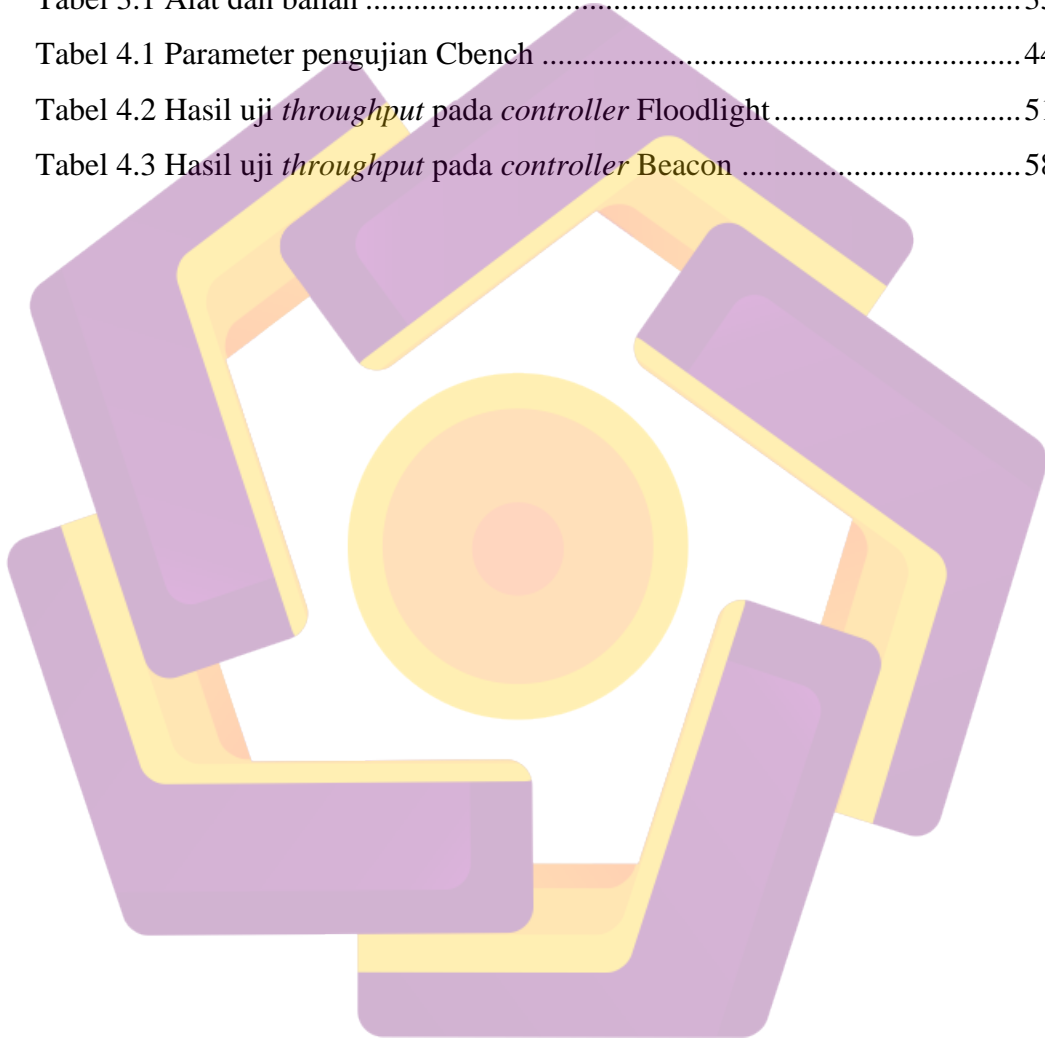
DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN	ii
HALAMAN PENGESAHAN	iii
PERNYATAAN	iv
MOTTO	v
PERSEMBAHAN	vi
KATA PENGANTAR	vii
DAFTAR ISI	viii
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
INTISARI	xiii
<i>ABSTRACT</i>	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	2
1.3 Batasan Masalah	2
1.4 Maksud dan Tujuan Penelitian	3
1.5 Metode Penelitian	3
1.5.1 Metode Pengumpulan Data	4
1.5.2 Metode Analisis Data	4
1.5.3 Metode Testing	5
1.6 Sistematika Penulisan	5
BAB II LANDASAN TEORI	6
2.1 Tinjauan Pustaka	6
2.2 Dasar Teori	9
2.2.1 Sejarah <i>Software-Defined Network</i>	9
2.2.2 Pengertian <i>Software-Defined Network</i>	10
2.2.3 <i>Control Plane</i> dan <i>Data Plane</i>	14

2.2.4	<i>Controller Software-Defined Networking (SDN)</i>	15
2.2.5	Protokol <i>OpenFlow</i>	17
2.2.6	<i>Controller Beacon</i>	20
2.2.7	<i>Controller Floodlight</i>	23
2.2.8	<i>Emulator Mininet</i>	25
2.2.9	<i>Cbench</i>	28
2.2.10	<i>Throughput dan Latency</i>	31
BAB III METODE PENELITIAN		33
3.1	Gambaran Umum.....	33
3.2	Alat dan Bahan.....	33
3.3	Alur Penelitian	34
3.4	Jenis Data	37
3.5	Variabel Pengujian.....	37
BAB IV HASIL DAN PEMBAHASAN		38
4.1	Instalasi Mininet	38
4.2	Instalasi <i>Controller Floodlight</i>	39
4.3	Instalasi <i>Controller Beacon</i>	40
4.4	Skenario Uji	41
4.4.1	Pengujian <i>throughput controller Floodlight</i>	44
4.4.2	Pengujian <i>throughput controller Beacon</i>	52
4.4.3	Perbandingan <i>throughput controller Floodlight dan Beacon</i>	59
BAB V PENUTUP.....		61
5.1	Kesimpulan	61
5.2	Saran	62
DAFTAR PUSTAKA		xv

DAFTAR TABEL

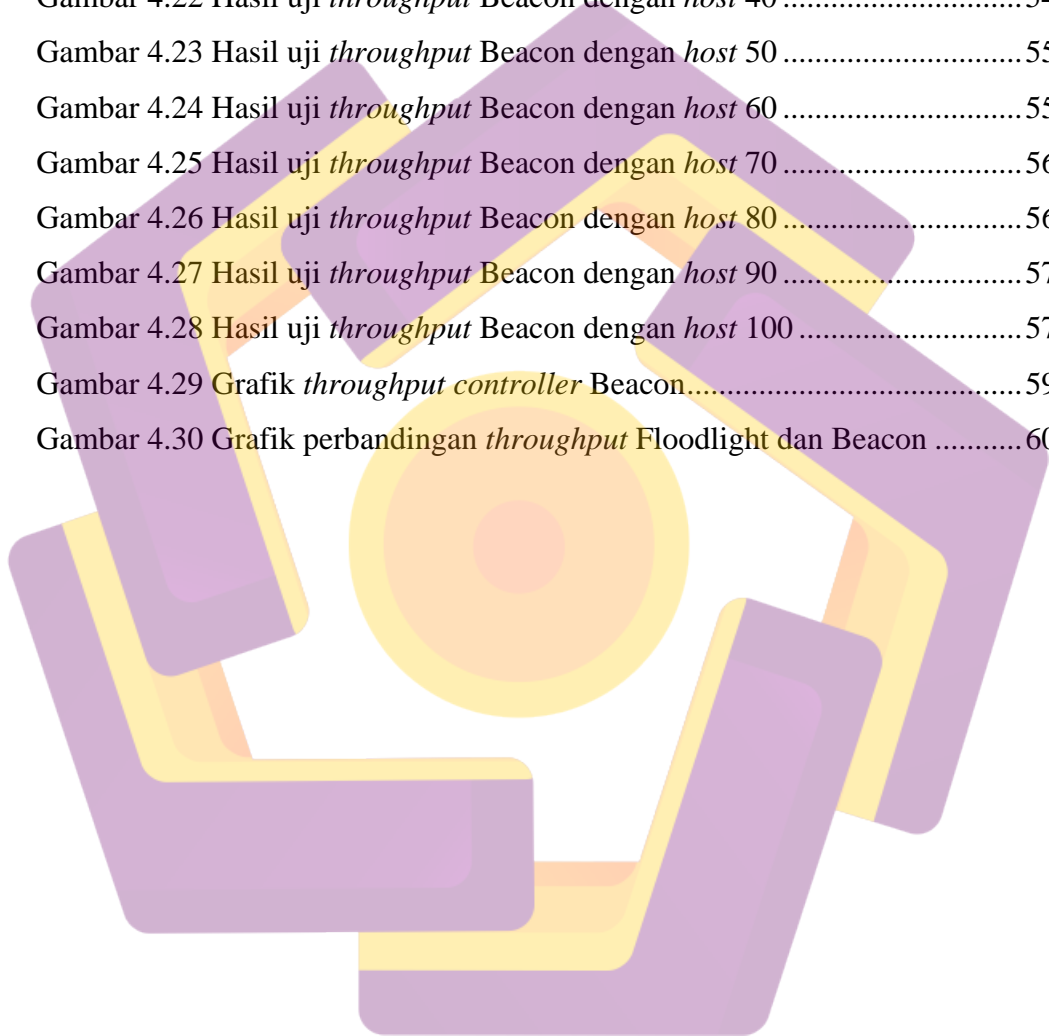
Tabel 2.1 Tabel Perbedaan Penelitian	8
Tabel 2.2 Perbedaan infrastruktur SDN <i>OpenFlow</i> dengan tradisional.....	13
Tabel 2.3 Opsi pengujian Cbench	31
Tabel 3.1 Alat dan bahan	33
Tabel 4.1 Parameter pengujian Cbench	44
Tabel 4.2 Hasil uji <i>throughput</i> pada <i>controller</i> Floodlight.....	51
Tabel 4.3 Hasil uji <i>throughput</i> pada <i>controller</i> Beacon	58



DAFTAR GAMBAR

Gambar 2.1 Perbedaan Jaringan Tradisional dan Jaringan SDN	11
Gambar 2.2 Layer <i>Software-Defined Networking</i>	14
Gambar 2.3 <i>Interface Controller</i> SDN	16
Gambar 2.4 Paket <i>flow</i> melalui proses <i>pipeline</i>	19
Gambar 2.5 Komponen utama tabel <i>flow</i>	20
Gambar 2.6 Beacon <i>IOFMessageListener</i>	22
Gambar 2.7 REST API <i>controller</i> Floodlight	24
Gambar 2.8 <i>Single Topology</i>	27
Gambar 2.9 <i>Tree Topology</i>	27
Gambar 2.10 <i>Linear Topology</i>	28
Gambar 2.11 Ilustrasi topologi yang dibentuk Cbench	29
Gambar 3.1 Alur Penelitian	36
Gambar 4.1 Install Mininet dalam VirtualBox	38
Gambar 4.2 GUI Floodlight	40
Gambar 4.3 GUI Beacon.....	41
Gambar 4.4 Perintah Membuat Topologi	42
Gambar 4.5 Hasil Perintah <i>Pingall</i>	42
Gambar 4.6 Perubahan GUI <i>Controller</i> Floodlight	43
Gambar 4.7 Perintah Uji <i>Throughput</i>	45
Gambar 4.8 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 10	46
Gambar 4.9 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 20	46
Gambar 4.10 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 30	47
Gambar 4.11 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 40	47
Gambar 4.12 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 50	48
Gambar 4.13 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 60	48
Gambar 4.14 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 70	49
Gambar 4.15 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 80	49
Gambar 4.16 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 90	50

Gambar 4.17 Hasil uji <i>throughput</i> Floodlight dengan <i>host</i> 100	50
Gambar 4.18 Grafik <i>throughput controller</i> Floodlight.....	52
Gambar 4.19 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 10	53
Gambar 4.20 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 20	53
Gambar 4.21 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 30	54
Gambar 4.22 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 40	54
Gambar 4.23 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 50	55
Gambar 4.24 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 60	55
Gambar 4.25 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 70	56
Gambar 4.26 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 80	56
Gambar 4.27 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 90	57
Gambar 4.28 Hasil uji <i>throughput</i> Beacon dengan <i>host</i> 100	57
Gambar 4.29 Grafik <i>throughput controller</i> Beacon.....	59
Gambar 4.30 Grafik perbandingan <i>throughput</i> Floodlight dan Beacon	60



INTISARI

Software-Defined Network (SDN), pertama kali diperkenalkan oleh Martin Casado di Universitas Stanford pada tahun 2007 dengan tulisan jurnalnya berjudul *Ethane: Taking Control of the Enterprise*. *Ethane* adalah sebuah penyederhanaan yang dari *ethernet switch* dengan sebuah *controller* terpusat yang mengatur hak masuk dan aliran *routing*. ONF (2012), mendefinisikan SDN adalah sebuah arsitektur jaringan baru dimana kontrol jaringan dipisahkan dari *forwarding* dan diprogram secara langsung. Selain itu ONF menyatakan bahwa dengan SDN tidak lagi membutuhkan protokol standar, tetapi cukup hanya menerima instruksi dari sebuah SDN *controller* dan menggunakan protokol *OpenFlow*. *Controller* SDN bersifat *open source* dan bermacam-macam, diantaranya POX, IRIS, MUL, Beacon, Floodlight, RYU, OpenDaylight, dan lain sebagainya. *Controller* merupakan bagian terpenting dalam jaringan SDN, karena bertindak sebagai kontrol terhadap lalu lintas data. Untuk mengetahui performa dari *controller* SDN perlu dilakukan sebuah pengujian.

Pada skripsi ini, peneliti mencoba menguji dua *controller* diantara semua *controller* SDN yang ada, yaitu Floodlight dan Beacon. Untuk mengetahui performa dari kedua *controller* tersebut, dilakukan pengujian *throughput*. Pengujian dilakukan dengan cara membandingkan hasil uji *throughput* menggunakan *tool benchmarking* Cbench.

Dari pengujian *throughput* pada kedua *controller* menghasilkan Floodlight sebagai *controller* yang stabil dengan terus meningkatnya nilai *throughput* seiring bertambahnya jumlah *host*. Sedangkan *controller* Beacon memiliki nilai *throughput* yang tidak stabil, pada jumlah *host* sebanyak 20, 50, dan 70 nilai *throughput* Beacon mengalami penurunan. Tetapi nilai *throughput* Beacon memiliki jumlah *flow* tiap detik lebih besar dari nilai *throughput* Floodlight. Pada Floodlight nilai *throughput* tertinggi sebesar 389,14 flow/s didapat ketika jumlah *host* sebanyak 100. Sedangkan Beacon nilai *throughput* tertinggi sebesar 1392,38 flow/s didapat ketika jumlah *host* sebanyak 10.

Kata-kunci: *throughput*, Floodlight, Beacon, *Software-Defined Network* (SDN)

ABSTRACT

Software-Defined Network (SDN), first introduced by Martin Casado at Stanford University in 2007 with a journal article titled Ethane: Taking Control of the Enterprise. Ethane is a simplification of ethernet switches with a centralized controller that regulates right of entry and routing flow. ONF (2012), defines arsitertur SDN is a new network which is separated from the network control and programmable forwarding directly. In addition ONF SDN stated that by no longer requiring a standard protocol, but quite simply receive instructions from a controller SDN and OpenFlow protocol. SDN Controller is open source and sundry, including POX, IRIS, MUL, Beacon, Floodlight, RYU, OpenDaylight, and so forth. Controller is an important part in the network SDN, because it acts as a control for data traffic. To determine the performance of the SDN controller needs to do a test.

In this thesis, the researcher tried to test two controller among all existing SDN controller, ie Floodlight and Beacon. To determine the performance of the second controller, testing throughput. Testing is done by comparing the results of throughput testing using benchmarking tool Cbench.

From the testing throughput on both the controller generates Floodlight as controller stable with increasing throughput with increasing the number of hosts. While the Beacon controller has a throughput value is not stable, the number of hosts as many as 20, 50, and 70 Beacon decreased throughput. But the throughput Beacon has a number of flow per second is greater than the value of throughput Floodlight. At Floodlight highest throughput value of 389.14 flow / s obtained when the number of hosts as many as 100. While Beacon highest throughput value of 1392.38 flow / s obtained when the number of hosts as many as 10.

Keywords: *throughput, Floodlight, Beacon, Software-Defined Network (SDN)*