

Prototipe Layanan VoIP Pada Jaringan OpenFlow

Eki Ahmad Zaki Hamidi, Mufid Ridlo Effendi, Hafizh Wibowo Widodo

Jurusan Teknik Elektro UIN Sunan Gunung Djati Bandung

Jl. AH. Nasution 105 Bandung - Indonesia

e-mail: ekiahmadzaki@uinsgd.ac.id, mufid.ridlo@uinsgd.ac.id, hafizhww@gmail.com.

Abstrak – *Voice Over Internet Protocol (VoIP)* adalah sebuah teknologi yang mampu melewatkan trafik suara dan data yang berbentuk paket melalui jaringan IP. Penggunaan IP memungkinkan penghematan biaya dikarenakan tidak perlu membuat sebuah infrastruktur baru untuk komunikasi suara. Dalam mengaplikasikan VoIP saat ini masih menggunakan infrastruktur jaringan konvensional, dengan berkembangnya SDN (*Software Defined Network*) yang menawarkan paradigma baru dalam dunia jaringan dalam mendesain, mengelola dan mengimplementasikan jaringan, terutama untuk mendukung kebutuhan dan inovasi di bidang ini yg semakin lama semakin kompleks, maka layanan VoIP dapat diaplikasikan pada jaringan OpenFlow yang merupakan implementasi dari konsep SDN (*Software Defined Network*). Dalam mengimplementasikan VoIP menggunakan pada jaringan OpenFlow, VoIP dapat dibangun dengan menggunakan aplikasi yang bersifat freeware seperti X-Lite sebagai User Agent atau Client, Asterisk sebagai server VoIP dan G.711 sebagai codec dengan memanfaatkan OpenvSwitch yang berfungsi meneruskan paket-paket dalam layanan VoIP. Pada pengujian ini dilakukan terbagi dalam 2 bagian, pengujian pada OpenvSwitch dan VoIP. Pada pengujian OpenvSwitch dapat disimpulkan bahwa flow yang sudah dibuat pada OpenvSwitch dapat berfungsi untuk meneruskan paket-paket dalam layanan VoIP sehingga antara server dan client dapat terhubung. Pengujian VoIP dilakukan sebanyak 10 kali percobaan dengan hasil yang didapatkan dengan rata-rata delay 10,0002421 ms, throughput 171,195 Kbps, jitter 0,424 ms, packet loss 0%

Kata Kunci: *Software Defined Network, OpenFlow, OpenvSwitch, VoIP*

1. Pendahuluan

VoIP (*Voice Over Internet Protocol*) adalah sebuah teknologi yang mampu melewatkan trafik suara, video dan data yang berbentuk paket melalui jaringan IP. Penggunaan jaringan IP memungkinkan penghematan biaya dikarenakan tidak perlu membuat infrastruktur baru untuk komunikasi suara dan penggunaan lebar data (*bandwidth*) yang lebih kecil dibandingkan dengan telepon biasa[1][2]. Saat ini ada tiga jenis metode berbeda dan yang paling sering digunakan dalam layanan VoIP yaitu ATA (*Analog Telephone Adapter*), *IP Phones*, *Computer to Computer*[3]. Infrastruktur yang ada saat ini selain menggunakan infrastruktur dengan metode konvensional sudah berkembang pula infrastruktur berbasis SDN (*Software Defined Network*).

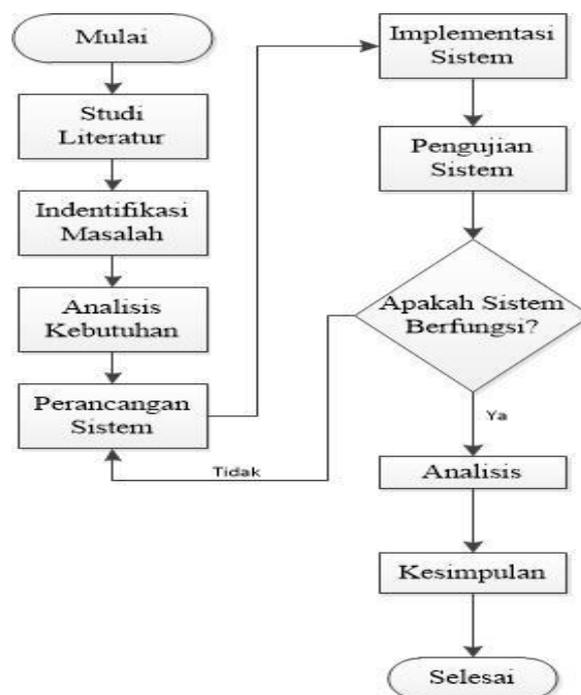
SDN erat kaitannya dengan *OpenFlow* sehingga banyak yang beranggapan bahwa SDN adalah *OpenFlow*. SDN lahir dari protokol *OpenFlow* yang diusulkan oleh Nick McKeown dan rekan-rekannya. *OpenFlow* adalah spesifikasi komunikasi antara *control plane* dan *data plane* *OpenFlow* merupakan standar terbuka yang diterapkan pada SDN[4][5]. Sebuah *switch OpenFlow* terdiri dari dua jenis yang pertama adalah *hardware-base switch*, *switch* jenis ini telah memodifikasi *hardware* dengan menggunakan sebuah OS khusus untuk mengimplementasikan *OpenFlow protocol* dan jenis kedua adalah *software base switch* yang

menggunakan system Unix atau Linux untuk mengimplementasikan seluruh fungsi *OpenFlow*[6].

Dengan mempertimbangkan fleksibilitas SDN (*Software Defined Network*) berbasis *OpenFlow* yang memungkinkan dapat diimplementasikan pada layanan VoIP (*Voice Over Internet Protocol*).

2. Metodologi Penelitian

Penelitian ini memiliki beberapa dengan tahapan. Tahap-tahap tersebut dipersiapkan untuk menunjang proses penelitian agar berjalan sistematis. Dari beberapa tahapan, kemudian disusun dalam bentuk diagram atau *flowchart* seperti berikut:



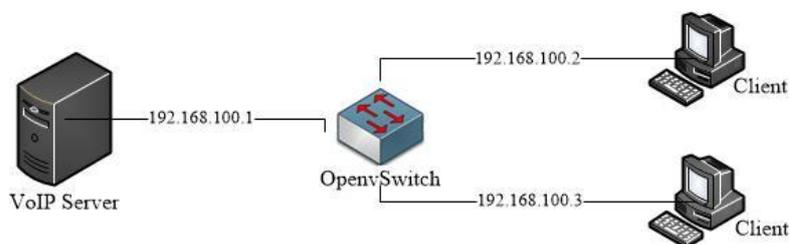
Gambar 1. Flowchart Metodologi Penelitian

3. Rancangan dan Implementasi

Perancangan yang digunakan adalah perancangan server VoIP menggunakan *software* Asterisk dengan Ubuntu 16.04 sebagai sistem operasinya. Selanjutnya untuk client VoIP menggunakan X-lite *Softphone* sehingga mampu melakukan penerimaan panggilan dengan sesama *client* yang sudah terdaftar pada server VoIP. Dan untuk mengimplementasikan layanan VoIP pada jaringan *OpenFlow* maka digunakan *OpenvSwitch* sebagai switch virtual *OpenFlow* dengan sistem operasi Linux Debian 8 dengan tambahan *network interface card* (NIC) yang berfungsi sebagai penghubung antara server dan client, kemudian dilakukan analisis pengujian dengan menggunakan *software* wireshark dengan parameter pengujian berupa *delay*, *jitter*, *packet loss* dan *throughput*.

3.1. Topologi Jaringan

Topologi yang digunakan adalah:



Gambar 2. Topologi Jaringan

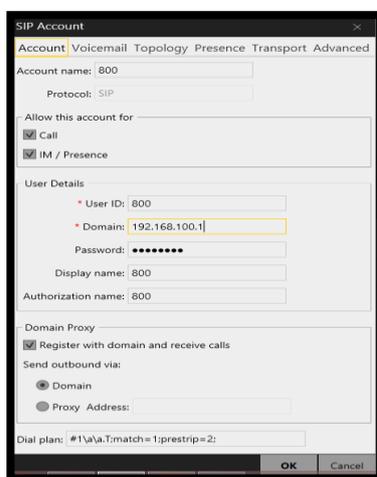
3.2. Konfigurasi Asterisk

Sebagian besar konfigurasi Asterisk terletak di direktori `/etc/asterisk`. File `SIP.conf` berisi parameter yang berkaitan dengan konfigurasi SIP untuk server Asterisk, dengan memberikan perintah “`nano /etc/asterisk/sip.conf`” yang berfungsi untuk membuat sebuah penomoran serta password pada client yang akan digunakan sehingga memungkinkan client dapat berkomunikasi dengan client lainnya.

Setelah proses konfigurasi pada file `SIP.conf` tahap selanjutnya adalah dengan melakukan konfigurasi pada file `extensions.conf` yang berfungsi untuk membuat pola panggilan yang akan dilakukan dengan menggunakan perintah “`nano /etc/asterisk/extensions.conf`”

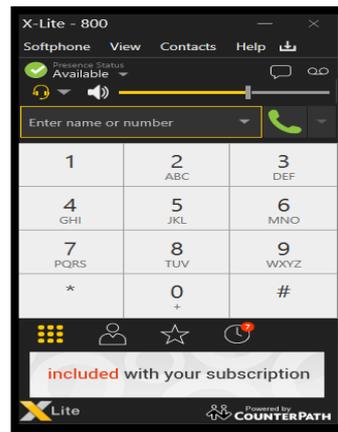
3.3. Konfigurasi Softphone

Beberapa konfigurasi sistem yang bertujuan agar sesama client yang telah terdaftar pada server VoIP dapat menerima dan melakukan panggilan.



Gambar 3. Konfigurasi Softphone

Dalam proses membuat akun yang sudah sesuai nama, nomor dan *password* maka akan secara otomatis *softphone* VoIP akan mendeteksi terhadap perubahan-perubahan parameter (syarat sudah terhubung dalam satu jaringan). Apabila tahap konfigurasi sudah tidak terdapat masalah maka akan terlihat *softphone* atau *client* VoIP yang sudah siap digunakan dalam menerima atau melakukan panggilan, dapat dilihat pada gambar 4 berikut dibawah ini:



Gambar 4. Softphone

3.4. OpenvSwitch

Switch berfungsi sebagai manajemen lalu lintas yang terdapat pada jaringan komputer, dan *switch* juga bertugas sebagai media untuk mengirimkan sebuah paket data untuk sampai ke tujuan dengan mencari jalur yang paling baik dan optimal serta memastikan pengiriman paket data ke tujuannya. *Switch* tersebut difungsikan sebagai *OpenvSwitch* atau *switch virtual OpenFlow* dan *switch* TP-link T1-SF11008D.

4. Hasil dan Analisa

Beberapa skenario analisa pada VoIP dengan tujuan mendapatkan sebuah hasil perbandingan antara penggunaan layanan VoIP menggunakan *switch OpenFlow* dan menggunakan *switch* TP-Link T1-SF1008D dan kemudian diambil nilai rata-rata dari setiap pengukuran.

4.1. Analisa Panggilan VoIP pada *Switch OpenFlow*

Proses pertama pada *OpenvSwitch* yang dilakukan adalah dengan *up-port* yang bertujuan agar port dalam *switch virtual* dalam keadaan aktif (Up) dan siap digunakan.

```
haf1zhww@haf1zhww-K435V:~$ sudo su
[sudo] password for haf1zhww:
root@haf1zhww-K435V:~/home/haf1zhww# ssh root@192.168.0.1
root@192.168.0.1's password:
Permission denied, please try again.
root@192.168.0.1's password:

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Apr  8 17:54:25 2017 from 192.168.0.2
root@debian:~# ifconfig ovs0 up
root@debian:~# ifconfig eth0 up
root@debian:~# ifconfig eth1 up
root@debian:~# ifconfig eth2 up
root@debian:~# ifconfig eth3 up
```

Gambar 5. Up Port *OpenvSwitch*

Proses yang kedua adalah dengan membuat *flow* pada *OpenvSwitch* dengan perintah `ovs-ofctl add-flow ovs0`

```

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Apr 8 17:54:25 2017 from 192.168.0.2
root@debian:~# ifconfig vsw0 up
root@debian:~# ifconfig eth0 up
root@debian:~# ifconfig eth1 up
root@debian:~# ifconfig eth2 up
root@debian:~# ifconfig eth3 up
root@debian:~# ovs-ofctl add-flow vsw0 arp,action-normal
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=3,ip,nw_dst=192.168.100.1,action=output:2"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=4,ip,nw_dst=192.168.100.1,action=output:2"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=2,ip,nw_dst=192.168.100.2,action=output:3"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=2,ip,nw_dst=192.168.100.3,action=output:4"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=3,ip,nw_dst=192.168.100.3,action=output:4"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=4,ip,nw_dst=192.168.100.2,action=output:3"
root@debian:~#

```

Gambar 6. Proses Add-flow OpenvSwitch

Proses selanjutnya adalah dengan melakukan *dump-flow* berfungsi untuk melihat paket yang digunakan, dan yang menjadi acuan pada *dump-flow* ini adalah *n_packet* dijadikan salah satu acuan yang dilihat karena *n_packet* muncul beberapa kali paket itu digunakan, pada pengujian ini *n_packet* yang didapat tidak 0 artinya *flow* sering digunakan.

```

192.168.1.1 - PUTTY
root@debian:~# ifconfig eth2 up
root@debian:~# ifconfig eth3 up
root@debian:~# ovs-ofctl add-flow vsw0 arp,action-normal
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=3,ip,nw_dst=192.168.100.1,action=output:2"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=4,ip,nw_dst=192.168.100.1,action=output:2"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=2,ip,nw_dst=192.168.100.2,action=output:3"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=2,ip,nw_dst=192.168.100.3,action=output:4"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=3,ip,nw_dst=192.168.100.2,action=output:3"
root@debian:~# ovs-ofctl add-flow vsw0 "priority=10,in_port=3,ip,nw_dst=192.168.100.3,action=output:4"
root@debian:~# ovs-ofctl dump-flows vsw0
NIST_FLOW reply (xid=0x4):
cookie=0x0, duration=58,478s, table=0, n_packets=2957, n_bytes=631646, idle_age=0, priority=10,ip,in_port=
2,nw_dst=192.168.100.3 actions=output:4
cookie=0x0, duration=101,950s, table=0, n_packets=0, n_bytes=0, idle_age=101, priority=10,ip,in_port=3,nw_
dst=192.168.100.1 actions=output:2
cookie=0x0, duration=19,534s, table=0, n_packets=0, n_bytes=0, idle_age=19, priority=10,ip,in_port=3,nw_ds
t=192.168.100.3 actions=output:4
cookie=0x0, duration=53,305s, table=0, n_packets=0, n_bytes=0, idle_age=53, priority=10,ip,in_port=2,nw_ds
t=192.168.100.2 actions=output:3
cookie=0x0, duration=86,644s, table=0, n_packets=0, n_bytes=0, idle_age=86, priority=10,ip,in_port=4,nw_ds
t=192.168.100.1 actions=output:2
cookie=0x0, duration=34,714s, table=0, n_packets=1754, n_bytes=374772, idle_age=0, priority=10,ip,in_port=
2,nw_dst=192.168.100.2 actions=output:3
cookie=0x0, duration=153,126s, table=0, n_packets=51, n_bytes=3060, idle_age=0, arp actions=NORMAL

```

Gambar 7. Proses Dump-flow OpenvSwitch

Proses *add-flow* pada *OpenvSwitch* sudah dilakukan maka proses selanjutnya adalah dengan memastikan bahwa koneksi jaringan pada server VoIP melalui *OpenvSwitch* terhadap *client* satu dan dua sudah terhubung dengan baik. Pengujian dilakukan dengan “ping” pada terminal dengan perintah “ping 192.168.100.2” untuk *client* satu dan “ping 192.168.100.3” untuk *client* dua.

```

root@hafizhww-K435V: /home/hafizhww
hafizhww@hafizhww-K435V:~$ sudo su
[sudo] password for hafizhww:
root@hafizhww-K435V: /home/hafizhww# ping 192.168.100.2
PING 192.168.100.2 (192.168.100.2) 56(84) bytes of data:
64 bytes from 192.168.100.2: icmp_seq=1 ttl=128 time=0.305 ms
64 bytes from 192.168.100.2: icmp_seq=2 ttl=128 time=0.357 ms
64 bytes from 192.168.100.2: icmp_seq=3 ttl=128 time=0.361 ms
64 bytes from 192.168.100.2: icmp_seq=4 ttl=128 time=0.362 ms
64 bytes from 192.168.100.2: icmp_seq=5 ttl=128 time=0.855 ms
64 bytes from 192.168.100.2: icmp_seq=6 ttl=128 time=0.341 ms
64 bytes from 192.168.100.2: icmp_seq=7 ttl=128 time=0.388 ms
^C
--- 192.168.100.2 ping statistics ---
7 packets transmitted, 7 received, 0% packet loss, time 5998ms
rtt min/avg/max/mdev = 0.305/0.424/0.855/0.177 ms
root@hafizhww-K435V: /home/hafizhww# ping 192.168.100.3
PING 192.168.100.3 (192.168.100.3) 56(84) bytes of data:
64 bytes from 192.168.100.3: icmp_seq=1 ttl=128 time=0.273 ms
64 bytes from 192.168.100.3: icmp_seq=2 ttl=128 time=0.272 ms
64 bytes from 192.168.100.3: icmp_seq=3 ttl=128 time=0.246 ms
64 bytes from 192.168.100.3: icmp_seq=4 ttl=128 time=0.261 ms
64 bytes from 192.168.100.3: icmp_seq=5 ttl=128 time=0.268 ms
^X64 bytes from 192.168.100.3: icmp_seq=6 ttl=128 time=0.318 ms
64 bytes from 192.168.100.3: icmp_seq=7 ttl=128 time=0.301 ms
64 bytes from 192.168.100.3: icmp_seq=8 ttl=128 time=0.268 ms
^C
--- 192.168.100.3 ping statistics ---
8 packets transmitted, 8 received, 0% packet loss, time 6996ms
rtt min/avg/max/mdev = 0.246/0.275/0.318/0.030 ms
root@hafizhww-K435V: /home/hafizhww#

```

Gambar 8. Cek Konektifitas

Dalam melakukan panggilan dalam VoIP setiap *client* harus sudah terdaftar pada *server* VoIP. Proses panggilan ini sangat mirip dengan komunikasi yang biasa dilakukan pada *handphone*, untuk melihat *client* VoIP yang aktif pada *server* adalah dengan memberikan perintah pada asterisk `-r sip show peers`.

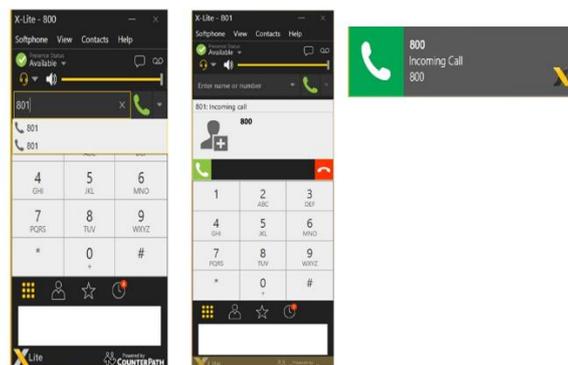
```

root@hafizhww-K435V:/home/hafizhww# asterisk -r
Privilege escalation protection disabled!
See https://wiki.asterisk.org/wiki/x/1gKFAQ for more details.
Asterisk 11.7.0-dfsg-1ubuntu1, Copyright (C) 1999 - 2013 Digium, Inc. and others

Created by Mark Spencer <markster@digium.com>
Asterisk comes with ABSOLUTELY NO WARRANTY; type 'core show warranty' for details.
This is free software, with components licensed under the GNU General Public
License version 2 and other licenses; you are welcome to redistribute it under
certain conditions. Type 'core show license' for details.
=====
Connected to Asterisk 11.7.0-dfsg-1ubuntu1 currently running on hafizhww-K435V (
pid = 2375)
hafizhww-K435V*CLI> sip show peers
Name/username          Host                                Dyn Forcerport
ACL Port              Status      Description
800/800                Unmonitored 192.168.100.2      D N
801/801                Unmonitored 192.168.100.3      D N
65508                  Unmonitored
    
```

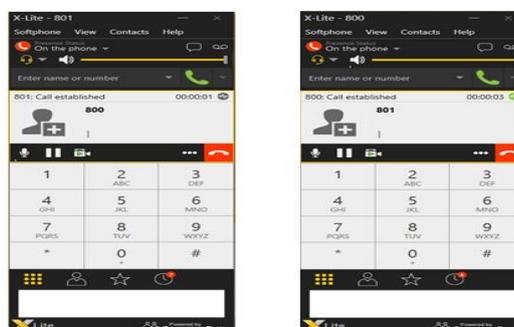
Gambar 9. Client VoIP sudah terdaftar pada server

Setelah akun *client* satu VoIP dengan nomor dial 800 dan akun *client* dua dengan nomor dial 801 sudah terdaftar pada *server* maka *client* VoIP dapat melakukan panggilan. Dengan menekan nomor dial yang sudah teregistrasi pada *server*, seperti pada gambar berikut ini:



Gambar 10. Client VoIP sudah terdaftar pada server

Pada gambar diatas adalah kondisi dimana *client* satu dengan nomor dial 800 akan melakukan panggilan dengan *client* dua yang mempunyai nomor dial 801. Pada proses tunggu ini, *client* dua terdapat tanda *incoming call* dari *client* satu yang sedang menunggu apakah permintaan call diterima atau ditolak. Pilih logo telepon untuk menerima komunikasi.



Gambar 11. Client VoIP on the phone

Apabila proses komunikasi berjalan (*client* dua menjawab telepon), maka informasi yang terdapat pada status *softphone client* satu dan dua berubah menjadi seperti gambar 5.8 diatas, informasi “*on the phone*” tersebut menandakan bahwa *client* satu dan *client* dua sedang beroperasi melakukan komunikasi.

4.2. Analisa Quality of Service

Adalah kemampuan suatu jaringan untuk menyediakan kemampuan suatu jaringan untuk menyediakan layanan yang lebih baik pada trafik tertentu pada berbagai jenis platform teknologi. Dan pada hasil penelitian VoIP yang didapat akan dibandingkan dengan rekomendasi ITU-T dan TIPHON mengenai standar kualitas layanan VoIP seperti dibawah ini :

- Delay (harus ≤ 150 ms, ITU-T G.114)
- Jitter (harus ≤ 75 ms, TIPHON)
- Packet Loss (harus $\leq 5\%$, ITU-T G.114)
- Throughput

Pengukuran dilakukan dengan penggunaan variasi switch yang digunakan. Switch yang digunakan untuk pengukuran adalah switch virtual *OpenFlow (OpenvSwitch)* dan *switch TP-Link T1-SF1008D*. pengambilan data dilakukan sebanyak 10 kali panggilan dengan waktu sekali panggilan adalah 5 menit. Pengukuran dilakukan dengan software wireshark dengan codec G.711.

4.2.1. Analisa Delay

Delay langsung berhubungan dengan kecepatan transfer data suatu jaringan, *delay* sangat berpengaruh terhadap data *real-time* seperti data suara pada aplikasi VoIP. Semakin besar nilai *delay* maka semakin buruk juga kualitas jaringan serta kualitas data yang diterima oleh pendengar [19]:



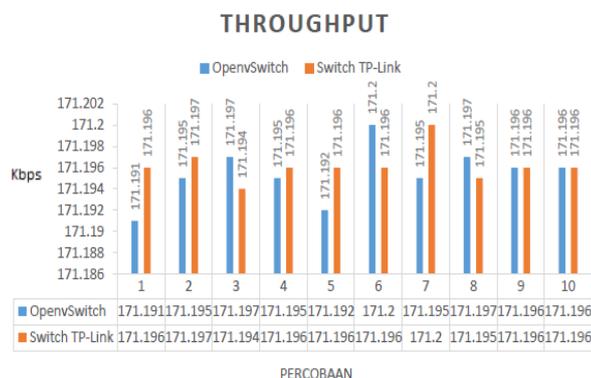
Gambar 12. Grafik Perbandingan Nilai Delay VoIP

Pada proses pengukuran *delay* pada komunikasi VoIP dengan menggunakan *OpenvSwitch* mempunyai nilai rata-rata *delay* sebesar 10.0002421 ms, sedangkan menggunakan *switch TP-Link* mempunyai nilai rata-rata *delay* sebesar 10,000989 ms. Dari hasil pengukuran kualitas suara pada layanan VoIP yang diterapkan *OpenvSwitch* dan pada *switch TP-Link* masih dianggap baik, dikarenakan menurut rekomendasi dari ITU-T dan TIPHON nilai *delay* yang disarankan adalah kurang dari 150 ms.

Dari hasil pengukuran besar nilai *delay* pada penelitian ini dipengaruhi oleh besar nilai *throughput*. Nilai *throughput* berbanding terbalik dengan nilai *delay*, semakin besar nilai *throughput* maka semakin kecil *delay* yang dihasilkan.

4.2.2. Data Throughput

Pengukuran throughput dilakukan untuk mengetahui laju data dalam suatu panggilan, sehingga bisa direkomendasikan persyaratan pada bandwidth minimal yang akan digunakan.

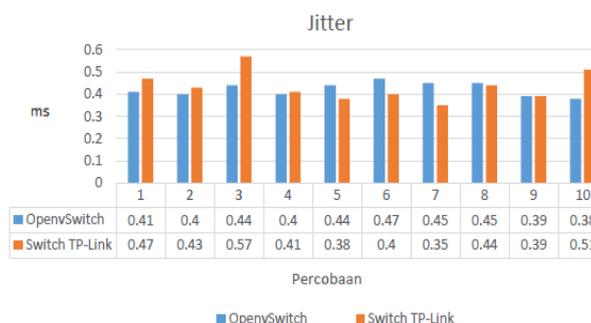


Gambar 13. Grafik Perbandingan Nilai Throughput VoIP

Pada pengukuran *throughput* dengan semakin lamanya durasi berbicara dengan menggunakan layanan VoIP, maka nilai *delay* yang ditimbulkan semakin besar, hal ini mempengaruhi nilai dari sebuah *throughput*. Dikarenakan pada pengujian ini dibatasi waktu pembicaraanya selama kurang lebih 5 menit, sehingga nilai rata-rata *throughput* pada *OpenvSwitch* sebesar 171,195 kbps, sedangkan menggunakan *switch* TP-Link mempunyai nilai rata-rata *throughput* sebesar 171,196 ms. Dikarenakan pada percobaan ini hanya menggunakan dua *client* dan masih berada pada jaringan lokal dan menggunakan media kabel sehingga nilai yang didapatkan pada penelitian ini relatif sama.

4.2.3. Analisa Jitter

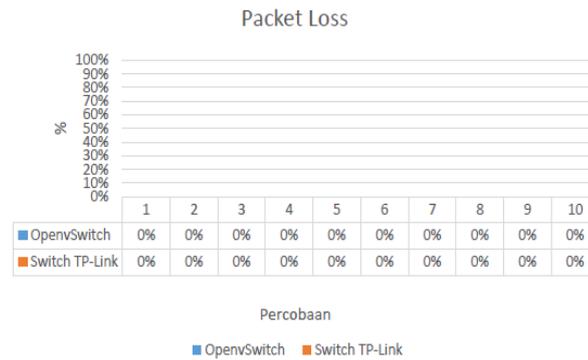
Jitter merupakan sebuah variasi *delay* antar paket yang terjadi pada jaringan IP, besarnya nilai *jitter* dipengaruhi oleh variasi beban trafik dan besarnya penumpukan antar paket yang ada dalam jaringan IP [19].



Gambar 14. Grafik Perbandingan Nilai Jitter VoIP

4.2.4. Analisa Packet Loss

Packet Loss adalah hilangnya satu atau lebih paket data yang berjalan pada jaringan atau dengan kata lain paket data yang gagal dalam mencapai tujuan [19].



Gambar 15. Grafik Perbandingan Nilai Packet Loss VoIP

Berdasarkan hasil pengukuran pada layanan VoIP dengan menggunakan *OpenvSwitch* dan *switch* TP-Link didapatkan hasil rata-rata adalah 0 % maka dari itu dapat dikategorikan adalah kondisi sangat bagus, karena pada standarisasi TIPHON mengkategorikan bahwa kualitas jaringan dengan nilai 0% adalah sangat bagus.

Berdasarkan pada hasil pengujian layanan VoIP (*Voice Over Internet Protocol*) yang sudah ditampilkan dalam bentuk grafik yang dilakukan dengan menggunakan *OpenvSwitch* dan *switch* TP-Link dengan beberapa skenario pengujian.

Dapat disimpulkan dengan menganalisa data-data yang dihasilkan oleh *software* wireshark tersebut, dari hasil pengujian yang sudah dilakukan dapat dilihat pada tabel berikut ini.

Tabel 1. Analisa Performansi Jaringan

| Quality of Service | OpenvSwitch | Switch TP-Link |
|--------------------|---------------|----------------|
| Delay | 10,0002421 ms | 10,0001989 ms |
| Throughput | 171,195 Kbps | 171,196 Kbps |
| Jitter | 0,424 ms | 0,435 ms |
| Packet Loss | 0% | 0% |

5. Kesimpulan

Kesimpulan Implementasi VoIP pada jaringan *OpenFlow* dengan menggunakan *OpenvSwitch* sebagai *switch virtual OpenFlow*. Dan sudah dilakukan sebuah pengukuran QoS terhadap sesi komunikasi dengan menggunakan SIP (*Session Innitiation Protocol*) untuk signaling, protokol RTP (*Real-Time Protocol*) untuk transport dan Codec audio G.711 maka dapat disimpulkan dari perancangan, implementasi dan pengujian sistem ini di dapat hasil sebagai berikut:

1. VoIP pada jaringan *OpenFlow* berhasil diimplementasikan.
2. Penggunaan priority pada *OpenvSwitch* pada pengujian ini tidak terlalu berpengaruh dikarenakan jaringan yang digunakan adalah skala kecil, namun pada skala besar akan berpengaruh.
3. *Port service* VoIP yang digunakan pada *OpenvSwitch* adalah random.
4. Pengujian VoIP dilakukan sebanyak 10 kali percobaan dengan hasil yang didapatkan dengan rata-rata *delay* 10,0002421 ms, *throughput* 171,195 Kbps, *jitter* 0,424 ms, *packet loss* 0%.

Daftar Pustaka

- [1] Lazuardi, N., 2009. Perencanaan Jaringan Komunikasi VoIP (*Voice Over Internet Protocol*) Menggunakan Asterisk SIP (*Session Initiation Protocol*).
- [2] Yuniati, Y., Fitriawan, H. and Patih, D.F.J., 2014. Analisa Perancangan Server VoIP (*Voice Over Internet Protocol*) dengan *Opensource* Asterisk dan VPN (*Virtual Private Network*) Sebagai Pengaman Jaringan Antar Client. *Jurnal Sains dan Teknologi Industri*, 12(1), pp.112-121.
- [3] Wahyuddin, M.I., 2009. Implementasi VoIP *Computer To Computer* Berbasis *Freeware* Menggunakan *Session Initiation Protocol*. , 3(1), pp.50–59.
- [4] Kartadie, R. and Satya, B., 2015. Uji Performa Implementasi *Software-Based OpenFlow Switch* Berbasis *OpenWRT* Pada Infrastruktur *Software-Defined Network*. *DASI*, 16(3), p.87.
- [5] Cui, Hongyan, et al. "Accurate Network Resource Allocation in SDN according to Traffic Demand." (2015). *openflow* halaman 26
- [6] Kartadie, R., Utami, E. and Pramono, E., 2014. Prototipe Infrastruktur *Software-Defined Network* Dengan Protokol *OpenFlow* Menggunakan UBUNTU Sebagai Kontroler. *DASI*, 15(1), p.24.
- [7] Gojali, I., 2013. Modul Jaringan Komputer.
- [8] Salwa, M.I., 2016. Implementasi DHCP server *multi subnet* menggunakan *switch OpenFlow*. Universitas Islam Negeri Sunan Gunung Djati Bandung.
- [9] Mahardwiani, I., 2015. Desain dan implementasi telepon internet pada jaringan komputer Laboratorium Telekomunikasi Politeknik Negeri Bandung. Politeknik Negeri Bandung.
- [10] Ilma, U.Z., 2011. Rancang bangun dan analisa *Quality of service (QoS)* pada sistem *Voice Over Internet Protocol (VoIP)* menggunakan *opensource* Elastix. Institut Sains dan Teknologi Nasional.
- [11] Supriyanto, 2013. Jaringan Dasar 1. Available at: <https://docs.google.com>
- [12] Syamsu, S., Modul Jaringan Komputer. Stimik AKBA
- [13] Li, Y., & Wang, G. (2013). *SDN-based switch implementation on network processors*. *Communications and Network*, 5(03), 434.
- [14] Tulloh, R., Negara, R.M. and Hidayat, A.N., 2015. Simulasi *Virtual Local Area Network (VLAN)* Berbasis *Software Defined Network (SDN)* Menggunakan POX Controller. *JURNAL INFOTEL*, 7(2), pp.129-136.
- [15] Baek, Sun Uk, et al. "Implementation and Verification of QoS Priority over Software Defined Networking." *Proceedings on the International Conference on Internet Computing (ICOMP)*. *The Steering Committee of The World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp)*, 2016
- [16] ONF Market Education Committee, 2012. *Software-defined networking: The new norm for networks*. *ONF White Paper*.
- [17] Risdianto, A.C., Arif, M. & Mulyana, E., 2013. Buku Komunitas SDN-RG: Pengantar SDN
- [18] Thorpe, Christina, et al. "iMOS: Enabling VoIP QoS Monitoring at Intermediate Nodes in an OpenFlow SDN." *Cloud Engineering Workshop (IC2EW), 2016 IEEE International Conference on. IEEE, 2016*.
- [19] Ketut Sudiarta, P. and Sukadarmika, G., 2012. Penerapan Teknologi VoIP Untuk Mengoptimalkan Penggunaan Jaringan Intranet Kampus Universitas Udayana. *Majalah Ilmiah Teknologi Elektro*, 8(2).