

Implementasi *Load Balancing* menggunakan Teknologi *EtherChannel* pada Jaringan LAN

KUKUH NUGROHO, MUHAMAD SYAMSUL FALLAH

Teknik Telekomunikasi IT Telkom Purwokerto
Email : kukuh@ittelkom-pwt.ac.id

Received 12 Juli 2018 | *Revised* 29 Agustus 2018 | *Accepted* 26 September 2018

ABSTRAK

Jumlah pengguna yang meningkat merupakan salah satu penyebab menurunnya kualitas jaringan jika hanya menggunakan ukuran bandwidth yang sama. Sebagai contoh pada implementasi jaringan kampus, jaringan LAN antar gedung yang berbeda biasanya dihubungkan dengan menggunakan perangkat switch. Penurunan performansi jaringan pada jaringan tersebut akibat penambahan jumlah pengguna bisa diatasi dengan menggunakan konsep load balancing dengan memanfaatkan teknologi EtherChannel. Implementasi teknologi tersebut dapat menggunakan dua pilihan protokol yaitu PAgP dan LACP. Penelitian ini bertujuan untuk menguji performansi jaringan yang menerapkan konsep load balancing dengan menggunakan protokol PAgP dan LACP. Skenario pengujian menggunakan konsep client-server, dimana aplikasi server yang digunakan adalah FTP untuk implementasi layanan pertukaran data dan penggunaan software VLC untuk implementasi layanan video streaming. Pada penggunaan layanan FTP dengan protokol PAgP dihasilkan nilai delay 42% lebih baik dibandingkan dengan LACP. Begitupula untuk nilai throughput, penggunaan protokol PAgP masih lebih baik 10% dibandingkan dengan LACP.

Kata kunci: LAN, Load balancing, Etherchannel, PAgP, LACP

ABSTRACT

The increasing number of users is one cause of declining network performance if only use on the same bandwidth value. For example, in the implementation on the campus network, LAN among different building is usually connected using a switch. The decreasing of network performance on that network due to the increasing of the number of users which can be overcome by using a load balancing concept with EtherChannel technology utilization. The implementation of that technology uses two protocols, namely PAgP and LACP. This research intends to test the network performance by applying the load balancing concept using both PAgP and LACP protocols. The experiment scenarios used a client-server concept, where the implemented application server is FTP for exchanging data services and VLC software for implementing video streaming services. In the use of FTP services with using PAgP protocol showed a delay value of 42% better than LACP. The same with throughput value, the use of PAgP protocol still better 10% than LACP.

Keywords: LAN, Load balancing, Etherchannel, PAgP, LACP

1. PENDAHULUAN

Penggunaan teknologi dalam jaringan komputer lebih difungsikan agar performansi jaringan yang digunakan untuk proses transfer data lebih meningkat (**Komputer, 2012**). Dalam jaringan LAN (*Local Area Network*) skala kecil dengan jumlah pengguna yang relatif sedikit, penggunaan *bandwidth* kecil mungkin tidak berpengaruh dengan tingkat kepuasan pengguna. Akan tetapi, seiring dengan pertambahan jumlah pengguna dalam jaringan LAN tersebut, maka perlu adanya proses pemetaan ulang jaringan, terutama dalam hal penggunaan *bandwidth*. Masalah ini sering dijumpai pada kebanyakan jaringan LAN, terutama pada implementasi jaringan kampus. Penambahan jumlah *student body* mahasiswa ikut menambah masalah dalam hal buruknya performansi jaringan apabila tidak diikutsertakan dengan penambahan jumlah *bandwidth*. Selain dengan cara meningkatkan ukuran *bandwidth*, salah satu solusi yang bisa digunakan adalah dengan menggunakan teknologi yang terdapat dalam perangkat *switch* yaitu *EtherChannel*. Konsep yang digunakan dalam teknologi *EtherChannel* adalah dengan menggabungkan beberapa jalur fisik menjadi satu jalur saja, akan tetapi sifatnya adalah *logic* (**Nugroho, 2017**).

Beberapa penelitian telah dilakukan untuk meneliti penggunaan teknologi *EtherChannel* maupun mengenai konsep *load balancing* yang disediakan oleh teknologi tersebut. Rohmat Tulloh (**Tulloh, 2017**) melakukan penelitian terhadap penggunaan teknologi *EtherChannel* namun diimplementasikan dalam jaringan SDN dengan menggunakan *controller* Ryu. Protokol yang digunakan adalah LACP. Dari hasil penelitian diperoleh data bahwa penggunaan teknologi *EtherChannel* memberikan perbaikan performansi jaringan dilihat dari parameter *throughput*, *jitter*, maupun *packet loss*. Misalnya jika dilihat dari sisi *throughput* diketahui bahwa penggunaan teknologi *EtherChannel* memberikan perbaikan sebesar 22,53% dibandingkan dengan konsep jaringan yang tidak menggunakan teknologi *EtherChannel*. Konsep topologi jaringan yang digunakan pada penelitian tersebut adalah dengan melakukan penggabungan sebanyak empat buah jalur fisik.

Konsep *link-aggregation* juga bisa diimplementasikan dengan menambahkan mekanisme keamanan. Penggabungan beberapa jalur yang digunakan untuk menghubungkan antar dua perangkat dapat dilakukan jika proses autentikasi antar kedua perangkat tersebut berhasil. Proses penggabungan jalur seperti ini, selain meningkatkan performansi jaringan juga dapat menambah sisi keamanan jaringan. Penelitian yang dilakukan oleh Amin (**Amin, 2014**) tentang implementasi konsep *link-aggregation* dengan menambahkan tingkat keamanan dan penggunaan konsep jaringan yang bersifat hirarki menunjukkan hasil bahwa penggunaan konsep tersebut masih dapat meningkatkan parameter nilai *throughput* jaringan. Studi kasus yang digunakan pada penelitian tersebut adalah jaringan yang terdapat pada STMIK dan Politeknik Palcomtech, Palembang. Metode penelitian yang digunakan adalah PPDIIO (*Prepare, Plan, Design, Implement, Operate, and Optimize*).

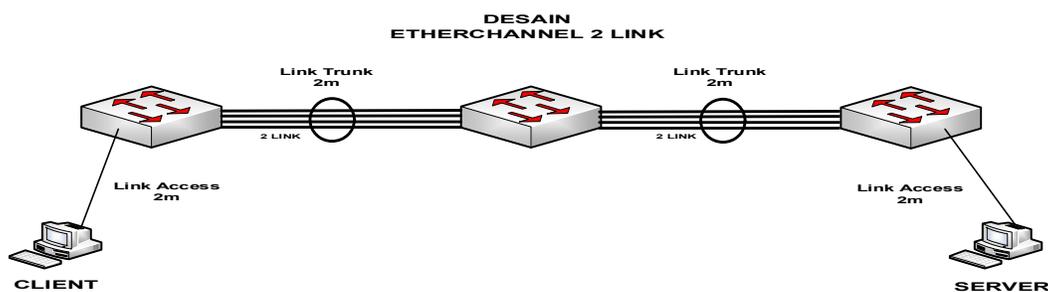
Perbaikan dari sisi performansi jaringan jika digunakan teknologi *EtherChannel* juga diperkuat oleh penelitian yang dilakukan oleh SysKconnect (**SysKconnect, 2002**) tentang konsep *link-aggregation* menggunakan standarisasi IEEE 802.3ad. Konsep *link-aggregation* adalah menggabungkan beberapa jalur menjadi satu jalur. Konsep ini sama hal dengan konsep penggunaan teknologi *EtherChannel*. Pada penelitian tersebut dinyatakan bahwa penggunaan konsep *link-aggregation* dapat menambah ukuran *bandwidth* saluran dan juga mampu menyediakan fasilitas *load balancing*.

Pada penelitian ini lebih difokuskan pada penggunaan teknologi *EtherChannel* dimana pilihan protokol yang digunakan adalah PAgP (*Port Aggregation Protocol*) dan LACP (*Link Aggregation Control Protocol*). Tujuan dari penggunaan kedua protokol tersebut adalah untuk

membandingkan performansi jaringan jika digunakan pilihan protokol PAgP buatan dari Cisco dan LACP yang dibuat oleh IEEE. Sifat penggunaan protokol PAgP adalah tertutup, artinya hanya perangkat produk dari Cisco saja yang bisa menggunakan protokol tersebut. Berbeda dengan penggunaan protokol LACP yang semua produk *switch* dari vendor manapun bisa menggunakan-nya. Manfaat dari penelitian ini adalah untuk mengetahui seberapa besar peningkatan performansi jaringan jika dimanfaatkan teknologi EtherChannel dengan menggunakan dua pilihan protokol yaitu PAgP dan LACP. Layanan yang digunakan dalam penelitian ini adalah FTP dan *video streaming*. Penggunaan kedua layanan tersebut tidak hanya diimplementasikan pada jaringan fisik, akan tetapi juga dengan menerapkan teknologi jaringan virtualisasi. Penelitian yang dilakukan oleh Djomi (**Djomi, 2018**) tentang penerapan teknologi jaringan virtualisasi dengan menggunakan teknologi *containers*. Layanan yang digunakan pada penelitian tersebut juga sama yaitu FTP dan *video streaming*.

2. METODOLOGI PENELITIAN

Metode penelitian yang digunakan adalah secara eksperimental di laboratorium. Tahapan awal penelitian dimulai dari proses studi literatur yang dilakukan dengan mencari referensi penelitian sebelumnya yang terkait dengan konsep *link-aggregation* atau *load balancing* melalui jurnal atau prosiding yang telah dilakukan sebelumnya. Dalam hal ini referensi yang berkaitan dengan penggunaan teknologi EtherChannel dengan menggunakan protokol PAgP (*Port Aggregation Protocol*) dan LACP (*Link Aggregation Control Protocol*). Tahapan selanjutnya adalah merancang topologi jaringan yang digunakan dalam proses penelitian. Dalam merancang topologi jaringan sebagai sistem yang diuji yaitu menggunakan konsep *link-aggregation*. Jumlah *link* atau jalur yang akan digabungkan pada saat proses penelitian menggunakan maksimal 8 jalur. Hal ini diperkuat setelah melakukan studi literatur tentang maksimal jalur yang bisa digabung oleh perangkat *switch* produk dari Cisco adalah sebanyak 8 jalur (**Community, 2007**).



Gambar 1. Desain topologi jaringan

Gambar 1 di atas menjelaskan topologi jaringan yang digunakan pada proses penelitian. Perangkat *switch* yang digunakan sebanyak tiga buah. Hal ini disesuaikan dengan ketersediaan jumlah perangkat *switch* Cisco (*Catalyst*) di laboratorium tempat uji jaringan. Antar *switch* dihubungkan dengan menggunakan kabel UTP cat6 dengan panjang 2 meter. Begitupula antara jarak *switch* dengan komputer, sehingga perkiraan jarak total antar kedua komputer *client* dan *server* adalah sebesar 8 meter. Dengan semakin panjang jarak kabel yang menghubungkan antar dua komputer, mengakibatkan terjadinya penurunan performansi jaringan, terutama terhadap peningkatan nilai *delay (latency)* dan adanya paket yang hilang (*packet loss*). Jarak maksimal panjang kabel UTP Cat6 yang dapat digunakan agar masih didapatkan nilai *packet loss* sebesar 0% adalah 256 meter (**Nugroho, K, 2017**). Walaupun penggunaan panjang kabel 8 meter saat pengujian jaringan tidak sebanding dengan panjang

kabel 256 meter, namun dengan menggunakan panjang kabel tersebut sudah dapat merepresentasikan akan adanya peningkatan nilai *delay* dibandingkan dengan hanya menggunakan panjang kabel kurang dari 8 meter. Penurunan performansi jaringan akibat jarak kabel penghubung dapat ditingkatkan dengan menggunakan metode penggabungan jalur (*link-aggregation*) antar dua perangkat *switch*. Sesuai dengan penjelasan Gambar 1 di atas, proses *load balancing* terjadi antar tiga perangkat *switch* yang terhubung. Jalur yang menghubungkan antar perangkat *switch* dinamakan sebagai jalur *trunk*. Konsep *link-aggregation* disini adalah dengan menggabungkan beberapa jalur *trunk* tersebut, dimana maksimal jalur yang digunakan adalah sebanyak 8 jalur. Walaupun penggunaan skenario uji jaringan sesuai dengan Gambar 1 di atas dapat meningkatkan performansi jaringan, namun jika dilihat dari sisi tingkat keamanan masih kurang. Data yang dipertukarkan antar dua komputer yang dihubungkan dengan hanya menggunakan perangkat *switch* akan rentang untuk diambil dan dibaca oleh pengguna yang lain. Hal ini dikarenakan kedua komputer tersebut masih terletak pada wilayah *broadcast* atau alamat *network* yang sama.

Perangkat yang digunakan terdiri dari perangkat keras (*hardware*) dan perangkat lunak (*software*). Perangkat keras yang digunakan pada penelitian ini diantaranya adalah 3 unit *switch* Cisco seri 2950 *Catalyst*, kabel UTP Cat5 sebagai media penghubung antara *switch* dan komputer, kabel *console* sebagai media untuk konfigurasi *switch* awal, serta 2 unit PC dengan spesifikasi Intel (R) Core(TM) i3-4005U CPU @1.70GHz 1.70GHz, RAM = 4GB. Adapun perangkat lunak yang digunakan pada penelitian ini diantaranya adalah *software Putty* untuk konfigurasi perangkat *switch*, USB merek Aten untuk menghubungkan antara *switch* dengan laptop, *FileZilla* untuk mengaktifkan server FTP, VLC media player untuk implementasi layanan *video streaming*, serta *software wireshark* untuk menghitung parameter performansi jaringan seperti *delay*, *packet loss*, *jitter*, dan *throughput*.

Konsep jaringan yang digunakan untuk proses uji performansi jaringan adalah *client-server*. Terdapat dua layanan yang digunakan yaitu file transfer dengan mengaktifkan *server* FTP dan *video streaming* dengan menggunakan bantuan *software* VLC *media player*. Nantinya PC client akan mengakses layanan yang diaktifkan di *server* baik itu FTP maupun *video streaming*. Pada saat yang bersamaan, *software wireshark* diaktifkan. Penggunaan *software* tersebut adalah untuk mengetahui nilai parameter pengukur dari performansi jaringan. Pengukuran performansi jaringan dilakukan ketika konsep *link-aggregation* sudah diaktifkan pada jalur *trunk*. Pilihan protokol yang digunakan pada proses penelitian ada dua yaitu PAgP dan LACP. Terdapat beberapa skenario pengukuran yaitu ketika hanya menggunakan dua jalur saja yang digabungkan, tiga jalur, empat jalur, lima jalur, enam jalur, tujuh jalur, sampai delapan jalur. Masing-masing skenario pengujian menghasilkan nilai parameter performansi jaringan yang berbeda-beda.

2.1 Parameter Performansi Jaringan

Pengukuran unjuk kerja jaringan dilakukan dengan menggunakan bantuan *software* *wireshark*. Terdapat empat parameter yang diamati saat proses pengujian jaringan yaitu *delay*, *jitter*, *packet loss*, dan *throughput*.

2.1.1 Delay

Delay adalah waktu yang dibutuhkan oleh sebuah paket untuk bisa sampai ke perangkat tujuan dari pertama kali bit pada paket tersebut dikirimkan oleh perangkat pengirim (Forouzan, 2013). Waktu tunda ini menyebabkan adanya keterlambatan paket yang dikirim untuk sampai pada penerima. Perhitungan untuk mengetahui nilai *delay* dari pengirim menuju penerima menggunakan Persamaan (1).

$$\text{Delay (sec)} = \frac{\text{Waktu antara paket 1 dan terakhir}}{\Sigma \text{ total paket}} \quad (1)$$

Pada saat *software wireshark* diaktifkan, *software* tersebut akan menangkap semua paket yang lewat dalam jaringan. Perhitungan nilai *delay* dilakukan hanya pada paket yang berasal dari komunikasi antara *client* dan *server*. Karena pada skenario topologi jaringan tidak menggunakan *background traffic*, maka paket yang beredar dalam jaringan hanyalah paket yang berasal dari kedua komputer tersebut. Persamaan (1) menjelaskan cara mendapatkan nilai *delay* dengan menggunakan *software wireshark* yaitu waktu antara pengiriman paket pertama sampai paket terakhir dibagi dengan jumlah paket yang diterima. Hasil dari perhitungan nilai *delay* akan dibandingkan dengan standarisasi nilai *delay* yang dikeluarkan oleh ITU-T G (ITU-T, 2003).

Tabel 1. Standarisasi nilai delay dari ITU-T G.114

<i>Delay</i> (<i>Latency</i>)	<i>Category</i>	<i>Delay</i>
<i>Standard</i>	<i>Good</i>	0-150 ms
	<i>Medium</i>	150-400 ms
	<i>Poor</i>	>400 ms

Tabel 1 menjelaskan standarisasi nilai *delay*. Hasil perhitungan nilai *delay* dikatakan baik apabila nilai-nya diantara 0 sampai 150 ms, sedangkan untuk kategori sedang yaitu antara 150 sampai 400 ms. Nilai tersebut dijadikan sebagai acuan untuk menilai hasil perhitungan saat proses pengujian jaringan.

2.1.2 Jitter

Jitter adalah variasi *delay* (Forouzan, 2013). Paket yang dikirimkan oleh perangkat pengirim akan sampai ke perangkat penerima dengan waktu kedatangan yang berbeda-beda. Rata-rata variasi waktu antar kedatangan paket dimanakan sebagai *jitter*. Pengertian dari *jitter* akan berhubungan dengan nilai *delay*. Variasi *delay* antar kedatangan atau pengiriman paket akan berpengaruh terdapat nilai *jitter*. Nilai *delay* besar disertai nilai *jitter* besar dapat dikatakan performansi jaringan dalam keadaan sangat buruk. Kondisi terbaik adalah jika didapatkan nilai *delay* dan *jitter* kecil. Hal ini menandakan performansi jaringan dalam keadaan sangat baik. Pengukuran nilai *jitter* menggunakan Persamaan (2).

$$\text{Jitter (sec)} = \frac{\Sigma \text{ delay-variasi}}{\Sigma \text{ paket terima-paket kirim}} \quad (2)$$

Hasil pengukuran nilai *jitter* akan dibandingkan dengan standarisasi *jitter* yang dikeluarkan oleh ITU-G.114 untuk mengetahui kualitas dari jaringan yang diamati.

Tabel 2. Standarisasi nilai jitter dari ITU-T G.114

<i>Jitter</i>	<i>Category</i>	<i>Jitter</i>
<i>Standard</i>	<i>Good</i>	0-20 ms
	<i>Medium</i>	20-50 ms
	<i>Poor</i>	>50 ms

Standarisasi nilai *jitter* masih lebih rendah dibandingkan nilai *delay*. Sebagai contoh untuk *jitter* kategori bagus terletak di antara rentang nilai 0 sampai 20 ms. Nilai jitter berada di rentang nilai sedang ada pada nilai 20 sampai 50 ms.

2.1.3 Throughput

Throughput merupakan sebuah ukuran yang menyatakan nilai *bandwidth* sebenarnya yang diterima oleh sebuah perangkat (**Forouzan, 2013**). Rumus untuk menghitung nilai *throughput* pada saat penelitian menggunakan Persamaan (4).

$$\text{Throughput} = \frac{\sum \text{ukuran paket kirim}}{\text{waktu pengiriman paket}} \text{ bps} \quad (4)$$

Dari Persamaan (4) nilai *throughput* merupakan hasil bagi dari jumlah total ukuran data yang dikirimkan dengan waktu total pengiriman data.

2.2 Implementasi Sistem

Pada penelitian ini digunakan perangkat *switch* asli yaitu tiga buah *Catalyst* seri 2950. Perangkat tersebut dihubungkan secara langsung seperti yang terlihat pada keterangan Gambar 1. Jalur *trunk* yang menghubungkan antar *switch* akan digabung dengan cara mengimplementasikan teknologi *EtherChannel*. Pilihan protokol yang digunakan adalah PAgP dan LACP.

Sebelum dilakukan proses konfigurasi, terlebih dahulu memastikan bahwa pada jalur *trunk*, kecepatan transfer data yang digunakan oleh interface Ethernet sama. Pada saat implementasi jaringan, tipe Ethernet yang digunakan pada jalur *trunk* adalah FastEthernet dengan kecepatan 100 Mbps. Hal ini dikarenakan salah satu syarat port *switch* bisa digabung adalah kecepatan transfer data antar port *switch* tersebut harus sama. Maksimal jalur saat melakukan proses pengujian adalah 8 jalur (port). Setelah memastikan kecepatan dari port *switch*, berikutnya adalah memastikan wilayah VLAN yang digunakan oleh masing-masing port adalah sama. Pada saat pengujian digunakan wilayah VLAN dengan nomer 1 (VLAN 1).

2.2.1 Konfigurasi PAgP

Dalam menggabungkan port *switch*, protokol PAgP akan melakukan proses negoisasi dengan port *switch* lawan. Proses tersebut dilakukan dengan cara mempertukarkan paket antar perangkat *switch* dan tipe paket tersebut tergantung dari protokol yang digunakan. Ketika protokol PAgP diaktifkan, pilihan berikutnya adalah menentukan mode dari port *switch* tersebut. Pilihan mode yang digunakan ada dua yaitu "*desirable*" dan "*auto*". Port *switch* yang diaktifkan mode "*desirable*" akan aktif dalam mengirimkan paket negoisasi ke perangkat *switch* lawan. Berbeda dengan mode "*auto*" yang memiliki sifat menunggu paket negoisasi dari perangkat *switch* lawan. Proses penggabungan port *switch* akan berhasil jika kombinasi mode yang digunakan benar. Misalnya memilih mode interface yang berbeda di kedua ujung yang saling terhubung yaitu mode "*desirable*" dan "*auto*".

```
Switch(config)#interface range fa0/1-3
Switch(config-if-range)#channel-protocol pagp
Switch(config-if-range)#channel-group 2 mode desirable
Switch(config-if-range)#
Creating a port-channel interface Port-channel 2
```

Gambar 2. Konfigurasi PAgP

Gambar 2 di atas menjelaskan cara mengkonfigurasi teknologi *EtherChannel* menggunakan protokol PAgP. Nomer grup yang digunakan pada saat pengujian yaitu 2. Grup tersebut menandakan kumpulan interface atau port yang telah digabungkan. Grup harus diberikan penanda sama seperti identitas port fisik *switch* seperti FastEthernet0/1. Setelah diaktifkan protokol PAgP kemudian disebutkan mode port *EtherChannel*. Mode port *switch* yang digunakan antara *switch* yang saling terhubung digunakan mode "*desirable*" dan "*auto*". Untuk

membuat mode "*auto*" perintah yang digunakan sama hanya mengubah mode "*desirable*" menjadi "*auto*".

2.2.2 Konfigurasi LACP

Protokol LACP menggunakan konsep yang sama seperti PAgP. Protokol tersebut akan menggabungkan port *switch* secara logic. Perbedaan hanya terletak pada mode port yang digunakan. Pilihan mode port yang digunakan oleh protokol LACP terdapat mode "*active*" dan "*passive*". Sebelum port *switch* yang telah digabungkan antara kedua *switch*, proses negoisasi harus dilakukan antara kedua *switch* tersebut.

```
Switch(config)#interface range fa0/1-3
Switch(config-if-range)#channel-protocol lacp
Switch(config-if-range)#channel-group 2 mode active
Switch(config-if-range)#
Creating a port-channel interface Port-channel 2
```

Gambar 3. Konfigurasi LACP

Konfigurasi *EtherChannel* menggunakan protokol LACP sama seperti dalam mengaktifkan protokol PAgP. Perbedaan utama hanya terletak pada penggunaan mode port-nya. Gambar 3 di atas menjelaskan hasil konfigurasi dimana mode yang digunakan adalah "*active*". Agar proses negoisasi berhasil dilakukan maka mode dari port lawan harus sama-sama "*active*" atau "*passive*". Terakhir dari langkah konfigurasi adalah mengaktifkan mode *trunk* pada kedua port *switch* hasil dari proses penggabungan antara beberapa port.

```
Switch(config)#interface port-channel 2
Switch(config-if)#switchport mode trunk
```

Gambar 4. Konfigurasi mode trunk

Gambar 4 menjelaskan cara mengaktifkan mode trunk pada interface *port-channel*. Interface tersebut merupakan interface logic dari dari beberapa interface fisik dalam perangkat *switch*.

2.2 Implementasi Sistem

Pengujian sistem dilakukan setelah menyiapkan skema topologi jaringan dan melakukan proses instalasi perangkat. Terdapat beberapa *software* yang digunakan untuk melakukan pengujian sistem seperti *Wireshark*, *VLC*, dan *FileZilla*. *Software Wireshark* digunakan untuk melakukan proses perhitungan nilai *delay*, *jitter*, *throughput*, dan *packet loss*. Pemasangan *software* tersebut ada pada komputer *client* dan *server*. Implementasi *server* dilakukan dengan menggunakan *software FileZilla* untuk proses transfer file dengan cara mengaktifkan *server* FTP. Sedangkan *software VLC* digunakan untuk mengimplementasikan skenario *video streaming*.

Skenario awal dilakukan dengan menjalankan aplikasi transfer file. Komputer *server* diaktifkan FTP di dalam *software* paket-an yang terdapat dalam *FileZilla*. Ukuran file yang dipertukarkan dibuat bervariasi dimulai dari 20 MByte, 30 MByte, 50 MByte, 70 MByte dan 100 MByte. Proses pengambilan data tidak bisa langsung didapatkan dari *software Wireshark*, akan tetapi hasil yang didapatkan dari *software* tersebut harus diubah ke dalam format ".csv" agar bisa dibaca oleh *software Microsoft Excel*. Hasil dari pengolahan dari *software* Ms. Excel dihasilkan empat parameter pengukur kinerja sistem yaitu *delay*, *jitter*, *packet loss*, dan *throughput*. Pengujian berikutnya yaitu dengan menjalankan aplikasi *video streaming*.

Sama seperti pengujian data yang sifat-nya *non-realtime* seperti pertukaran *file* dengan menggunakan protokol FTP, pengujian *video streaming* juga menggunakan acuan ukuran dari *file* video yang dipertukarkan antara komputer *client* dan *server*. Terdapat lima ukuran *file* video yang digunakan dalam proses pengujian yaitu 20 MByte, 30 MByte, 50 MByte, 70 MByte

dan 100 Mbyte, sama seperti pengujian dengan menggunakan layanan transfer file. *Software* untuk melakukan uji sistem masih sama yaitu *Wireshark*. Pengamatan dilakukan dengan menggunakan perintah "`udp.port==1234`" pada *software Wireshark* untuk mendapatkan data hasil *video streaming*.



Gambar 5. Proses *filtering* di Wireshark

No.	Time	Source	Destination	Protocol
2070	46.603289	172.16.0.254	192.168.20.10	MPEG TS
2071	46.604168	172.16.0.254	192.168.20.10	MPEG-1
2211	48.542420	172.16.0.254	192.168.20.10	MPEG TS
2212	48.543574	172.16.0.254	192.168.20.10	MPEG TS
2213	48.543580	172.16.0.254	192.168.20.10	MPEG TS
2214	48.547962	172.16.0.254	192.168.20.10	MPEG TS
2215	48.547970	172.16.0.254	192.168.20.10	MPEG TS
2216	48.547974	172.16.0.254	192.168.20.10	MPEG TS
2217	48.547977	172.16.0.254	192.168.20.10	MPEG TS
2218	48.547980	172.16.0.254	192.168.20.10	MPEG TS
2219	48.547983	172.16.0.254	192.168.20.10	MPEG TS
2220	48.548854	172.16.0.254	192.168.20.10	MPEG TS
2221	48.549758	172.16.0.254	192.168.20.10	MPEG TS
2222	48.549764	172.16.0.254	192.168.20.10	MPEG TS
2223	48.551207	172.16.0.254	192.168.20.10	MPEG TS
2224	48.551215	172.16.0.254	192.168.20.10	MPEG TS
2225	48.553654	172.16.0.254	192.168.20.10	MPEG TS
2226	48.553662	172.16.0.254	192.168.20.10	MPEG TS
2227	48.553665	172.16.0.254	192.168.20.10	MPEG TS

Gambar 6. Hasil *filtering* `udp.port == 1234`

Penggunaan perintah *filtering* pada Gambar 5 menghasilkan proses komunikasi hanya antara komputer *client* dan *server* saat dijalankan aplikasi *video streaming* seperti yang terlihat pada keterangan Gambar 6. Di sisi *server* diaktifkan *software* VLC untuk menjalankan layanan *video streaming*, sehingga untuk menampilkan komunikasi antara *client* dengan *server* yang diaktifkan *software* tersebut digunakan nomor port 1234 dengan protokol layer *transport* yang digunakan adalah UDP (*User Datagram Protocol*). Penggunaan layanan *video streaming* nantinya digunakan sebagai data pembandingan dari hasil uji sistem terhadap data yang sifatnya *non-realtime* seperti proses transfer *file*.

3. HASIL PENGUJIAN

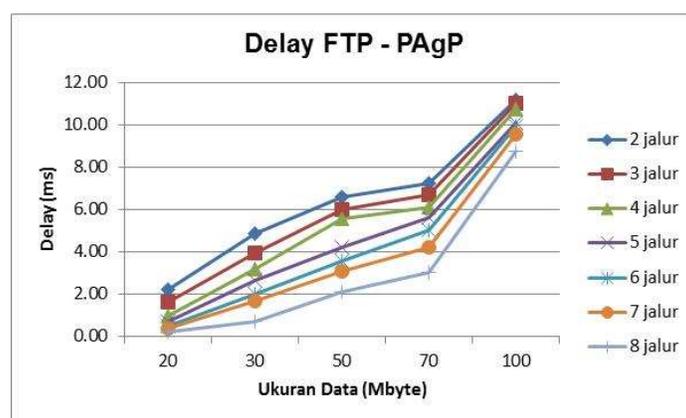
Topologi jaringan yang digunakan sebagai sistem uji terlihat pada keterangan Gambar 1. Pada topologi jaringan tersebut akan digunakan dua pilihan protokol yaitu PAgP dan LACP sebagai protokol untuk mengimplementasikan teknologi *EtherChannel*. Masing-masing protokol akan diujikan pada skema topologi jaringan seperti yang terlihat pada Gambar 1. Perbedaan yang mendasar dari skema dasar topologi jaringan yang digunakan untuk proses uji sistem adalah pada penggunaan jalur yang akan digabungkan. Jumlah jalur yang digabungkan dimulai dari penggunaan 2, 3, 4, 5, 6, 7, dan 8 jalur. Masing-masing jalur akan dilihat performansi jaringan-nya dengan menggunakan pilihan protokol yang berbeda. Terdapat tiga parameter yang digunakan untuk melihat performansi jaringan yaitu *delay*, *jitter*, dan *throughput*. Sistem diuji dengan mengalirkan trafik data, dimana layanan yang digunakan adalah berupa proses transfer data dan *video streaming*.

3.1 Pengujian Parameter *Delay* dan *Jitter*

3.1.1 Layanan Transfer Data

Pada pengujian ini masih menggunakan skema topologi jaringan pada Gambar 1. Jaringan akan diuji dengan melakukan pembebanan berupa proses transfer data. Di sisi *server* diaktifkan *software* FileZilla yang digunakan untuk mengaktifkan aplikasi *server* FTP. Kemudian pada *server* tersebut diberikan file dengan ukuran yang bervariasi mulai dari 20 Mbyte, 30 Mbyte, 50 Mbyte, 70 Mbyte, dan 100 Mbyte untuk nantinya *file* tersebut akan diakses oleh komputer *client*. Pada masing-masing proses transfer *file* menggunakan ukuran yang berbeda-beda. Kemudian topologi jaringan yang sudah diterapkan mekanisme *link-aggregation* akan diuji menggunakan parameter *delay*, *jitter*, dan *throughput*.

Parameter awal yang diuji adalah *delay*. Proses transfer *file* dilakukan antara komputer *client* dan *server*. Waktu proses pengiriman data dari komputer *server* sampai ke *client* akan dihitung. Skenario proses pertukaran data menggunakan ukuran *file* yang berbeda-beda. *File* tersebut tersimpan dalam *server* FTP, kemudian diunduh oleh komputer *client*. Pengujian dilakukan dengan menggunakan jumlah *link* yang digabung berbeda-beda, dimulai dari 2 jalur, 3 jalur, sampai 8 jalur.

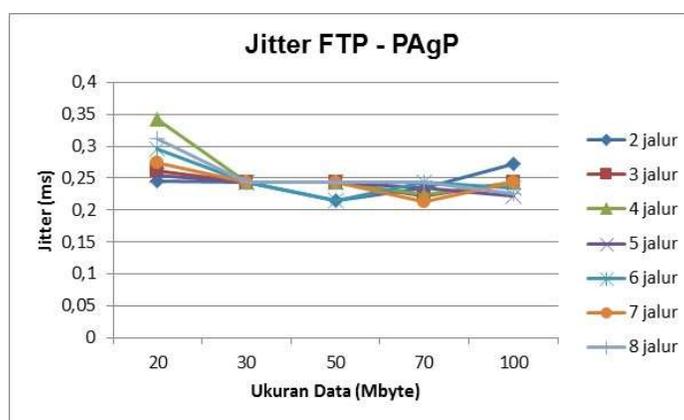


Gambar 7. Delay layanan transfer file PAgP

Proses pengambilan data dimulai dari penggabungan 2 jalur. Proses penggabungan jalur tersebut dilakukan pada jalur *trunk*. Gambar 7 di atas memperlihatkan hasil pengujian nilai *delay* saat layanan yang digunakan adalah berupa proses pertukaran data (*file*) dari komputer *server* ke *client*. Nilai *delay* terkecil terlihat jika jumlah jalur yang digabungkan maksimal yaitu 8 jalur dan yang terbesar yaitu pada penggunaan 2 jalur. Hal ini terlihat bahwa dengan menggabungkan jumlah jalur yang semakin banyak, nilai *delay* yang dihasilkan juga semakin kecil. Dengan menggabungkan jalur, beban trafik data akan dibagi, sehingga waktu yang dibutuhkan untuk mengirimkan paket agar bisa sampai ke tujuan juga semakin kecil. Dari penjelasan grafik pada gambar 7 terlihat juga bahwa dengan memperbesar ukuran paket yang dipertukarkan antara *client* dan *server* FTP, nilai *delay* yang dihasilkan juga semakin besar.

Pengaruh penggunaan teknologi EtherChannel sangat berpengaruh terhadap perbaikan performansi jaringan. Hal ini bisa dilihat dari hasil keluaran uji parameter *delay* dengan menggunakan topologi jaringan pada keterangan Gambar 7 di atas. Misalnya pada proses pertukaran data dengan menggunakan ukuran 20 Mbyte. Pada saat menggunakan jumlah jalur yang digabung sebanyak 2 jalur diperoleh nilai *delay* sebesar 2,23 ms. Nilai tersebut masih dikatakan baik menurut standar yang dikeluarkan oleh ITU-T G.114. Kemudian ketika jumlah jalur yang digabung ditambah menjadi 3 jalur, terdapat penurunan nilai *delay* sebesar 1,62

ms. Perbaikan nilai *delay* dengan menambah jumlah jalur yang digabung pada kasus ini adalah sebesar 38%. Ketika jumlah jalur yang digabung ditambah menjadi 4 jalur, juga terjadi penurunan nilai *delay* yaitu 1,01 ms. Artinya terdapat perbaikan performansi jaringan sebesar 60%. Dari grafik yang terlihat pada keterangan Gambar 7, nilai *delay* akan menurun dengan menambahkan jumlah jalur yang digabung. Seperti contoh pada penggunaan ukuran data 20 Mbyte. Dengan menggunakan ukuran data tersebut diketahui bahwa rata-rata prosentase peningkatan performansi jaringan sekitar 47%. Begitupula untuk hasil uji dengan menggunakan ukuran data yang lain. Dengan memperbesar ukuran dari data yang digunakan, dihasilkan penurunan nilai dari prosentase perbaikan performansi jaringan. Nilai rata-rata prosentase perbaikan performansi jaringan berturut-turut untuk penggunaan ukuran data sebesar 30 Mbyte, 50 Mbyte, 70 Mbyte, dan 100 Mbyte adalah 42%, 21%, 16%, dan 4%. Hal ini menandakan bahwa dengan semakin besar ukuran data yang digunakan, nilai kenaikan prosentase perbaikan performansi jaringan juga semakin kecil.

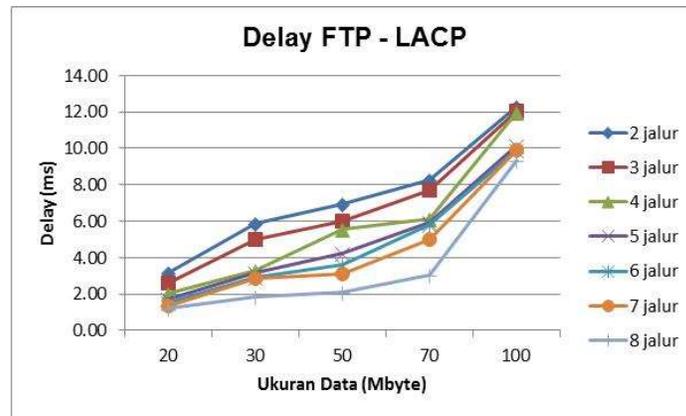


Gambar 8. Jitter layanan transfer file PAgP

Nilai *delay* akan berkaitan dengan nilai *jitter*. Waktu paket bisa sampai ke perangkat penerima berhubungan dengan interval waktu antar kedatangan paket. Gambar 8 menjelaskan hasil dari pengujian nilai *jitter*. Dari grafik yang ditunjukkan pada Gambar 8 terlihat bahwa nilai *jitter* relatif sama walaupun dengan memperbesar ukuran data atau menambah jumlah jalur yang digabung. Secara konsep dengan menambah jumlah jalur, maka dimungkinkan nilai *jitter* akan menurun. Hal ini dikarenakan adanya peningkatan kecepatan transfer data dengan menambah jumlah jalur. Namun berdasarkan keterangan topologi jaringan pada Gambar 1, *bandwidth* yang diberikan oleh jaringan *trunk* dengan menggunakan konsep *link-aggregation* tidak sesuai dengan *bandwidth* yang tersedia pada jalur akses yaitu hubungan antara perangkat *switch* dengan komputer. Walaupun jumlah *link* (jalur) pada jalur trunk ditambah, namun jumlah *link* pada jalur akses tetap satu jalur, dimana tipe Ethernet yang digunakan pada perangkat komputer dan *switch* adalah FastEthernet dengan ukuran *bandwidth* secara teori adalah 100 Mbps.

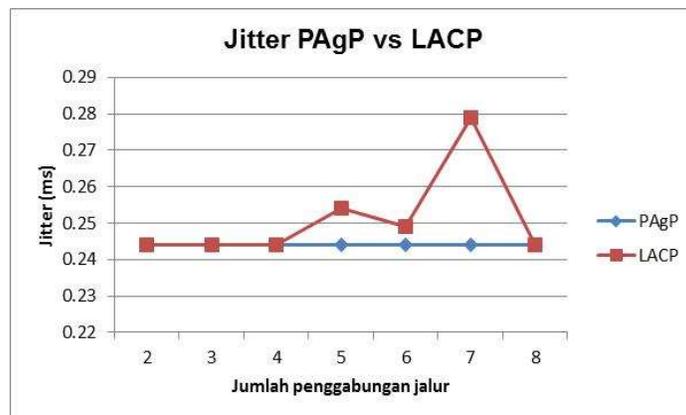
Dengan memperbesar ukuran data, jumlah paket yang dikirimkan juga semakin besar. Hal ini dikarenakan adanya proses segmentasi sebelum data bisa dikirimkan lewat jaringan. Dengan menggunakan teori ini, dapat disimpulkan bahwa nilai *jitter* tidak dipengaruhi oleh perubahan ukuran data, namun dapat dipengaruhi oleh penggunaan konsep *link-aggregation*. Gambar 8 menjelaskan bahwa dengan memperbesar ukuran data, nilai *jitter* yang dihasilkan relatif stabil. Misalnya pada penggunaan 3 jalur untuk proses *link-aggregation* diperoleh nilai *jitter* rata-rata yaitu sebesar 0,243 ms. Gambar 8 juga menjelaskan adanya penurunan nilai *jitter* disaat digunakan ukuran data 70 Mbyte, namun dengan selisih perubahan nilai yang tidak terlalu besar dengan penggunaan ukuran data 50 Mbyte yaitu 0,022 ms. Masalah ini dikarenakan

adanya ketidaksesuaian ukuran *bandwidth* antara jalur *trunk* dengan akses. Sehingga menyebabkan adanya *delay processing* pada perangkat *switch*.



Gambar 9. Delay layanan transfer file LACP

Gambar 9 di atas merupakan hasil pengukuran nilai *delay* pada saat digunakan protokol LACP. Tren grafik perubahan ukuran *file* terhadap nilai *delay* juga sama seperti penggunaan protokol PAgP, namun dengan nilai yang berbeda. Sebagai contoh pada penggunaan ukuran *file* 20 Mbyte dan jalur yang digabungkan sebanyak 8 jalur. Pada penggunaan protokol PAgP, nilai *delay* yang dihasilkan adalah 2,232 ms. Sedangkan pada penggunaan protokol LACP sebesar 3,173 ms. Hal ini terlihat bahwa penggunaan protokol LACP menghasilkan nilai *delay* yang relatif lebih besar dibandingkan penggunaan protokol PAgP. Begitupula ketika besaran data yang dipertukarkan sebesar 100 Byte. Penggunaan protokol PAgP menghasilkan nilai *delay* 11,176 ms, sedangkan LACP menghasilkan nilai *delay* sebesar 12,262 ms. Hasil pengukuran nilai *delay* tersebut apabila dilihat dari tabel standarisasi yang dikeluarkan oleh ITU-T masih dalam kategori baik (0 - 150 ms).



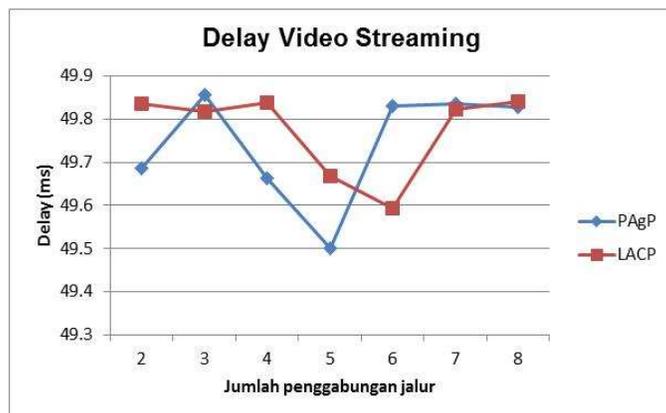
Gambar 10. Perbandingan rata-rata jitter PAgP dan LACP

Perbandingan nilai *jitter* ketika digunakan protokol PAgP dengan LACP dijelaskan pada keterangan Gambar 10. Grafik nilai *jitter* yang ditampilkan pada keterangan gambar tersebut merupakan nilai *jitter* dari hasil proses pengujian dengan menggunakan ukuran data sebesar 30 Mbyte dan jumlah pengujian sebanyak tiga puluh kali percobaan. Data grafik dari hasil pengujian memperlihatkan bahwa penggunaan protokol LACP menghasilkan nilai *jitter* yang relatif lebih besar dibandingkan ketika menggunakan protokol PAgP. Perbedaan nilai *jitter* mulai terlihat ketika digunakan jumlah jalur yang digabungkan sebesar 5 jalur. Penggunaan

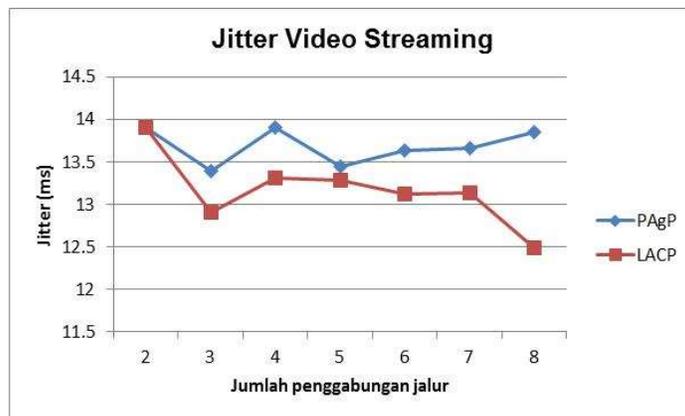
protokol PAgP menghasilkan nilai *jitter* sebesar 0,244 ms, sedangkan nilai *jitter* disaat menggunakan protokol LACP sebesar 0,254 ms. Perbedaan nilai tersebut relatif kecil. Perbedaan deviasi nilai *jitter* terbesar terjadi pada saat digunakan 7 jalur yang digabung. PAgP menghasilkan nilai *jitter* sebesar 0,244 ms, sedangkan LACP menghasilkan nilai yang lebih besar yaitu 0,279 ms. Grafik nilai *jitter* ketika digunakan protokol PAgP relatif konstan yaitu di angka 0,244 ms. Artinya pada pengujian sebanyak tiga puluh kali percobaan dihasilkan nilai rata-rata *jitter* yaitu sebesar 0,244 ms. Berbeda dengan nilai rata-rata *jitter* ketika digunakan protokol LACP yaitu 0,251 ms. Nilai *jitter* ketika digunakan protokol PAgP cenderung konstan, sedangkan pada penggunaan protokol LACP cenderung tidak stabil. Namun perbedaan nilai *jitter* tersebut relatif tidak terlalu besar. Ketidakstabilan nilai *jitter* disaat digunakan protokol LACP lebih disebabkan karena ketidaksesuaian antara protokol dengan perangkat uji yang digunakan. LACP adalah protokol yang dibuat oleh IEEE, sedangkan PAgP adalah protokol yang dibuat oleh Cisco. Pada saat melakukan proses pengujian, perangkat *switch* yang digunakan adalah produk dari Cisco. Dari hasil pengujian nilai *jitter* untuk layanan FTP terlihat bahwa *switch* produk dari Cisco hanya cocok dengan protokol produk dari Cisco untuk proses *link-aggregation* yaitu PAgP.

3.1.2 Layanan *Video Streaming*

Layanan berikutnya yang diujikan adalah *video streaming*. Pengambilan data *streaming* dimulai dari komputer *client* ke *server* yang memanfaatkan *software* VLC. Pengukuran nilai *delay* masih menggunakan *software* Wireshark. Pengambilan data dilakukan pada komputer *client*. File *video* berukuran 70 Mbyte dijalankan pada komputer *server*, kemudian diakses oleh komputer *client*.



Gambar 11. Perbandingan *delay* layanan *video streaming*



Gambar 12. Perbandingan *jitter* layanan *video streaming*

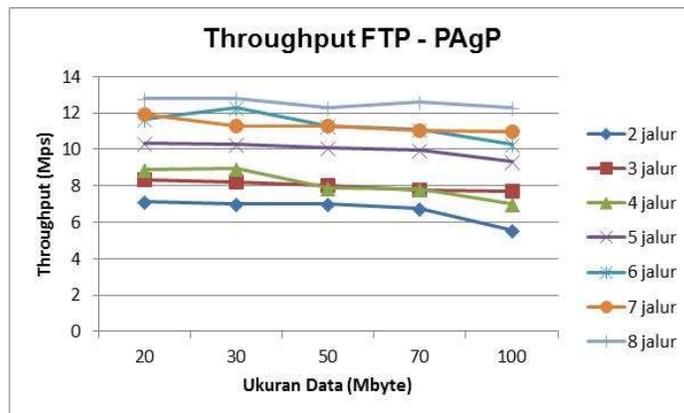
Pengujian *delay* dan *jitter* untuk layanan *video streaming* hanya dilakukan pada satu ukuran data yaitu 70 Mbyte. Gambar 11 di atas menjelaskan hasil pengujian *delay* ketika digunakan protokol PAgP dan LACP. Jumlah jalur yang digabungkan dimulai dari 2 jalur, 3 jalur, sampai 8 jalur. Pada pengujian menggunakan dua jalur, penggunaan protokol PAgP menghasilkan nilai *delay* yang lebih kecil dibandingkan dengan LACP. Nilai *delay* yang dihasilkan oleh PAgP sebesar 49,685 ms, sedangkan LACP menghasilkan nilai *delay* sebesar 49,834 ms. Pada pengujian berikutnya dengan menggunakan 3 jalur yang digabungkan, nilai *delay* LACP sedikit lebih kecil dibandingkan dengan nilai *delay* dari PAgP. LACP menghasilkan nilai *delay* 49,857 ms, sedangkan nilai *delay* yang dihasilkan oleh PAgP sebesar 49,818 ms. Pada pengujian kedua dengan menggunakan 3 jalur yang digabung tersebut masih belum diketahui tren grafik yang menunjukkan apakah penggunaan protokol PAgP lebih baik dari LACP begitupula sebaliknya. Namun pada pengujian ketiga dan keempat dengan menggunakan 4 dan 5 jalur yang digabung diketahui adanya tren grafik yang menunjukkan bahwa penggunaan protokol PAgP menghasilkan nilai *delay* yang lebih kecil dibandingkan dengan LACP. Tren grafik mulai terlihat pada penggunaan 3 sampai 5 jalur yang digabung yaitu cenderung menurun. Akan tetapi pada penggunaan protokol PAgP, tren grafik mulai naik saat digunakan 6 jalur yang digabung. Berbeda dengan menggunakan protokol LACP yang masih dalam tren menurun dan mulai naik saat digunakan 7 jalur yang digabung. Hal ini memperlihatkan adanya anomali tren grafik hasil pengujian dari keterangan Gambar 11. Penyebab utama kejadian tersebut adalah adanya ketidaksesuaian nilai *bandwidth* antara jalur utama, dalam hal ini adalah jalur *trunk* dengan jalur akses yang menghubungkan perangkat *switch* dengan komputer. Nilai *bandwidth* yang digunakan pada jalur akses adalah sebesar 100 Mbps dengan menggunakan tipe FastEthernet, sedangkan kalau jumlah jalur yang digabung sebesar 5 jalur, maka akan menghasilkan nilai *bandwidth* pada jalur *trunk* sebesar 5 x 100 Mbps (500 Mbps). Hal ini terjadi perbedaan nilai *bandwidth* yang sangat besar antara jalur *trunk* dengan jalur akses. Akibat kejadian ini maka terjadi penumpukan jumlah antrian paket pada perangkat *switch* sebelum paket tersebut dikirimkan ke komputer penerima lewat jalur akses. Dalam jaringan komputer istilah ini dinamakan dengan *bottleneck*. Perbedaan nilai *bandwidth* yang besar antara jalur *trunk* dengan jalur akses menyebabkan adanya *delay processing* pada perangkat *switch*. *Delay processing* juga akan menambah nilai *delay end-to-end* antara perangkat komputer pengirim dan penerima.

Dari penjelasan grafik pada Gambar 11 memperlihatkan bahwa adanya batasan optimal penggunaan jumlah jalur yang digabung jika dilihat dari sisi parameter *delay*. Misalnya pada penggunaan protokol PAgP, jumlah optimal jalur yang digabung adalah sebesar 5 jalur, sedangkan pada penggunaan protokol LACP adalah sebesar 6 jalur. Gambar 11 juga menjelaskan adanya perbedaan hasil nilai *delay* antara penggunaan protokol PAgP dan LACP dilihat dari sisi peningkatan performansi jaringan. Pada saat digunakan protokol PAgP, nilai *delay* cenderung mulai menurun pada saat digunakan penggabungan 3 jalur dan mulai kembali naik saat digunakan 6 jalur yang digabung. Berbeda dengan penggunaan protokol LACP. Nilai *delay* mulai turun disaat digunakan 4 jalur yang digabung dan mulai naik saat digunakan 7 jalur yang digabung. Adanya perbedaan nilai ini memperlihatkan respon perbaikan performansi jaringan protokol PAgP lebih baik dibandingkan LACP. Namun demikian hasil pengujian terlihat berbeda pada penggunaan parameter *jitter* seperti yang terlihat keterangan grafik pada Gambar 12. Pengujian dengan menggunakan 2 jalur, baik pada penggunaan protokol PAgP maupun LACP menghasilkan nilai *jitter* yang sama yaitu 13,907 ms. Namun apabila dilihat pada keseluruhan hasil pengujian dengan jumlah jalur yang digabungkan berbeda-beda, terlihat bahwa nilai *jitter* pada saat menggunakan protokol LACP masih lebih kecil dibandingkan jika digunakan protokol PAgP.

3.2 Pengujian Parameter *Throughput*

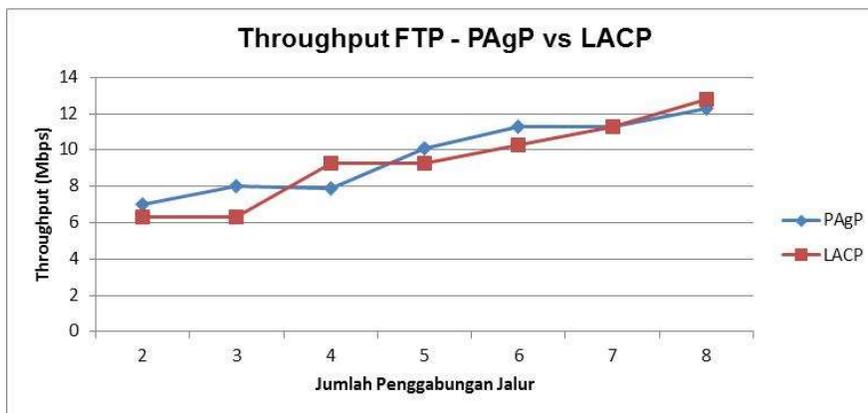
3.2.1 Layanan File Transfer

Selain *delay*, parameter lain yang diujikan adalah *throughput*. Perhitungan nilai *throughput* masih menggunakan *software* wireshark. Jumlah total ukuran dari paket yang dikirim dibagi dengan waktu pengiriman paket. Hal ini sesuai pada penjelasan perhitungan nilai *throughput* pada Persamaan (4). Layanan awal yang digunakan adalah proses transfer *file* dengan memanfaatkan protokol FTP.



Gambar 13. Hasil perhitungan nilai *throughput* FTP

Proses transfer *file* dalam jaringan digunakan protokol PAgP. Jumlah jalur yang digabungkan sama seperti pada saat melakukan pengujian dengan menggunakan parameter *delay*. Penggabungan jalur dimulai dari 2 jalur sampai 8 jalur. Sesuai pada keterangan Gambar 13, nilai *throughput* akan semakin besar dengan semakin banyak jumlah jalur yang digabungkan. Dari hasil pengujian terlihat bahwa penggabungan jumlah jalur sangat memberikan dampak yang positif dalam hal performansi jaringan. Konsep penggabungan jalur yang dilakukan oleh protokol PAgP terbukti dapat meningkatkan nilai *throughput* bagi *end-user*. Misalnya pada saat penggunaan ukuran data 20 Mbyte. Penggunaan 2 jalur dalam melakukan proses *load balancing* memberikan nilai *throughput* sebesar 7,253 Mbps. Sedangkan ketika jalur yang digabungkan menjadi 8 jalur, nilai *throughput* yang didapatkan oleh komputer *client* sebesar 11,179 Mbps.



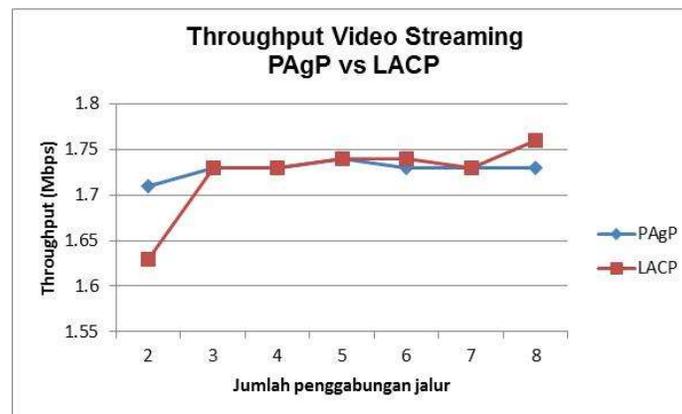
Gambar 14. Perbandingan nilai *throughput* PAgP dan LACP

Perhitungan nilai *throughput* juga bisa dibandingkan hasilnya ketika protokol yang digunakan adalah LACP. Gambar 14 di atas menjelaskan perbandingan hasil pengukuran nilai *throughput*,

dimana ukuran data yang digunakan hanya 50 Mbyte. Proses pengujian tidak dilakukan pada semua ukuran data. Hasil pengujian memperlihatkan bahwa tidak adanya kestabilan nilai. Grafik nilai *throughput* pada keterangan Gambar 14 memperlihatkan penggunaan protokol PAgP memberikan nilai *throughput* yang lebih baik dibandingkan jika menggunakan LACP. Hal ini bisa dilihat pada penggunaan jalur yang digabungkan sebanyak 2, 3, 5, 6, dan 7 jalur. Namun pada penggunaan penggabungan jalur tertentu terdapat hasil yang berbeda. Pada penggabungan 4 dan 8 jalur, penggunaan protokol LACP menghasilkan nilai *throughput* yang relatif lebih baik jika dibandingkan dengan PAgP. Namun dari tren grafik perbandingan nilai *throughput* pada keterangan Gambar 14 memperlihatkan bahwa penggunaan protokol PAgP masih memberikan nilai *throughput* yang lebih baik dibandingkan dengan penggunaan protokol LACP.

3.2.2 Layanan Video Streaming

Pengukuran parameter *throughput* dengan menggunakan layanan *video streaming* masih dilakukan pada sisi komputer *client*. Terdapat file yang berisi *video* diakses oleh komputer *client*. Ukuran resolusi dari *video* yang digunakan dalam proses pengujian adalah sebesar 360p. Agar komputer *client* bisa mengakses file *video* dari *server*, *software* VLC diaktifkan baik pada sisi komputer *client* maupun *server*.



Gambar 15. Hasil nilai *throughput* untuk layanan *video streaming*

Gambar 15 menjelaskan hasil perbandingan nilai *throughput* ketika jaringan diaktifkan teknologi EtherChannel dengan menggunakan protokol PAgP dan LACP. Ukuran file dari video yang ditelakkan di sisi komputer *server* adalah sebesar 70 Mbyte. File tersebut kemudian diakses oleh komputer *client* dengan menggunakan bantuan software VLC. Software *Wireshark* juga diaktifkan pada sisi komputer *client* untuk menghitung nilai *throughput*. Perhitungan nilai *throughput* sendiri menggunakan Persamaan (4).

Grafik yang ditunjukkan pada keterangan Gambar 15 di atas memperlihatkan bahwa belum terjadi konsistensi nilai antara ketika digunakan protokol PAgP maupun LACP. Pengujian dilakukan dengan menambahkan jumlah jalur yang digabungkan pada jalur *trunk*. Pada pengujian dengan menggunakan 2 jalur, nilai *throughput* ketika digunakan protokol PAgP menghasilkan nilai yang lebih besar dibandingkan dengan ketika digunakan protokol LACP. Pada pada pengujian dengan menggunakan 3, 4, 5, 6, dan 7 jalur dihasilkan nilai *throughput* yang relatif sama yaitu dengan rata-rata nilai *throughput* sebesar 1,73 Mbps. Namun jika dilihat dari tren hasil nilai *throughput* yang dihasilkan pada keterangan Gambar 15 di atas terlihat bahwa penggunaan protokol PAgP menghasilkan nilai *throughput* dengan tren nilai yang relatif lebih stabil dibandingkan dengan ketika digunakan protokol LACP yaitu di angka 1,73 Mbps. Sehingga jika dilihat dari tingkat kestabilan nilai *throughput* untuk penggunaan

layanan *video streaming*, penggunaan protokol PAgP masih lebih baik dibandingkan dengan penggunaan protokol LACP.

4. KESIMPULAN

Berdasarkan hasil pengujian jaringan dengan memanfaatkan teknologi EtherChannel dengan menggunakan protokol PAgP dan LACP diperoleh hasil bahwa penggunaan teknologi tersebut mampu meningkatkan performansi jaringan jika dilihat dari parameter *delay*, *jitter*, dan *throughput*. Pada implementasi layanan FTP, penggunaan protokol PAgP masih lebih baik dibandingkan dengan protokol LACP, baik dilihat dari parameter *delay*, *jitter*, maupun *throughput*. Namun berbeda hasilnya pada penggunaan layanan *video streaming*. Penggunaan protokol PAgP memberikan performansi jaringan yang lebih baik dibandingkan dengan LACP hanya dilihat dari sisi parameter *delay* dan *throughput*, namun hasil pengujian berbeda ketika digunakan parameter *jitter*. Sehingga dari hasil pengujian dapat disimpulkan bahwa penggunaan teknologi EtherChannel pada perangkat *switch* produk dari Cisco hanya cocok menggunakan protokol buatan Cisco yaitu PAgP. Penggunaan protokol tersebut mampu memberikan peningkatan performansi jaringan yang lebih baik dibandingkan jika digunakan protokol LACP. Namun perlu adanya penelitian lebih lanjut jika perangkat *switch* yang digunakan adalah bukan produk dari Cisco.

DAFTAR RUJUKAN

- Komputer, W. (2012). *Optimalisasi Jaringan Komputer Kabel & Nirkabel*. Yogyakarta: Andi Publisher.
- Nugroho, K. (2017). *Switch & Multilayer Switch Cisco*. Bandung: Informatika.
- Tulloh, R. (2017). Analisis Performansi Agregasi Link Dengan LACP Pada SDN Menggunakan Ryu Sebagai Controller. *Jurnal Nasional Teknik Elektro*, 6(3), 203-213.
- Amin, Z. (2014). Simulasi dan Perancangan Keamanan Autentikasi Jaringan Hirarki Link Aggregation Control Protocol (LACP) Berbasis Router Cisco (Studi Kasus: STMIK PALCOMTECH). *Konfrensi Nasional Teknologi Informasi dan Aplikasinya*, (pp. 9-17).
- SysKonnct. (2002). *Link Aggregation according to IEEE Standard 802.3ad*. White Paper.
- Djomi, M. M. (2018). Analisis Performansi Layanan FTP dan Video Streaming berbasis Network Function Virtualization menggunakan Docker Containers. *ELKOMIKA*, 6(2), 180-193.
- Community, C. S. (2007, July 9). *Understanding EtherChannel Load Balancing and Redundancy on Catalyst Switches*. Dikutip dari <https://www.cisco.com/c/en/us/support/docs/lan-switching/etherchannel/12023-4.html>
- Nugroho, K. (2017). Uji Performansi Jaringan Menggunakan Kabel UTP dan STP. *ELKOMIKA*, 5(1), 48-59.
- Forouzan, B. A. (2013). *Data Communications and Networking 5E, Fifth Edition*. United States: McGraw-Hill.
- ITU-T. (2003, May). *Series G: Transmission Systems and Media, Digital Systems and Networks (One-way transmission time)*. Dikutip dari <https://www.itu.int/rec/T-REC-G.114/en>.