

Pengaruh *Native VLAN* terhadap *Overhead* Trafik Data pada Jaringan LAN

KUKUH NUGROHO

Institut Teknologi Telkom Purwokerto
Email: kukuh@ittelkom-pwt.ac.id

Received 6 Desember 2020 | *Revised* 15 Desember 2020 | *Accepted* 2 Januari 2021

ABSTRAK

Penggunaan switch pada jaringan yang sudah menerapkan teknologi VLAN dapat membagi jaringan menjadi beberapa wilayah jaringan yang berbeda dimana proses ini sebelumnya hanya bisa dilakukan oleh router. Data yang dipertukarkan antar wilayah jaringan yang berbeda akan melewati jalur trunk yang menghubungkan antar perangkat switch. Proses penambahan header ke dalam data dilakukan perangkat switch sebelum diteruskan melewati jalur trunk. Namun proses tersebut akan menambah overhead (beban trafik tambahan) dari setiap data yang dilewatkan pada jalur trunk. Produk switch dari Cisco memiliki fasilitas yang dinamakan sebagai native VLAN. Apabila asal wilayah VLAN dari data yang dilewatkan pada jalur trunk sama dengan informasi native VLAN yang tersimpan pada perangkat switch, maka data tersebut tidak perlu ditambahkan header disaat ingin dilewatkan jalur trunk. Penelitian ini bertujuan untuk menguji pengaruh penggunaan native VLAN dalam mengurangi beban trafik jaringan pada jalur trunk. Hasil pengujian memperlihatkan penggunaan native VLAN dapat mengurangi beban overhead trafik data sebesar 4,77%.

Kata kunci: VLAN, Native VLAN, Trunk, Jaringan LAN, Overhead Trafik

ABSTRACT

The use of switches in the switch network with implementing VLAN technology can divide a network into several different networks that previously just be done with a router. The data that is exchanged among different networks will pass the trunk link that connects between switches. The process of adding a header within data is done by a switch before entering that data into the trunk link. However, that process will increase overhead traffic (additional traffic load) within each data that passed on the trunk link. Cisco's switches have a facility known as native VLAN. If the origin of the data's VLAN area that will be passed on the trunk link is the same as native VLAN information, then the data will not add with a header when will pass on the trunk link. This research will analyze the effect of using native VLAN in reducing overload data traffic on the trunk link. The test result shows that the use of native VLAN can reduce data traffic overhead by 4.77%.

Keywords: VLANs, Native VLANs, Trunk Link, Switch Network, Overhead Traffic

1. PENDAHULUAN

Ethernet adalah protokol yang paling banyak digunakan dalam membangun jaringan LAN (*Local Area Network*) (Khalaf, dkk, 2015). Cakupan jaringan LAN terbatas dalam sebuah wilayah, misalnya jaringan pada sebuah kantor atau kampus. Jaringan LAN yang menghubungkan antar gedung yang berbeda dan perangkat penghubung yang digunakan relatif banyak dikarenakan kebutuhan pengguna dengan layanan yang beragam memerlukan konsep *load balancing* untuk menjaga performansi jaringan lebih optimal, salah satu caranya adalah dengan menggunakan teknologi *EtherChannel* (Nugroho & Fallah, 2018). Implementasi jaringan LAN dapat menggunakan media kabel atau *wireless* yang menghubungkan antar perangkat. Trafik data akan dibangkitkan antar perangkat dengan hubungan linier, artinya semakin besar jumlah perangkat dalam jaringan, ukuran trafik data yang beredar dalam jaringan juga semakin besar. Pengaturan trafik data perlu dilakukan agar didapatkan performansi jaringan yang optimal. Jaringan LAN pada umumnya akan menggunakan switch sebagai perangkat penghubung antar komputer perlu juga dilakukan proses optimasi jaringan agar performansi jaringan LAN tersebut sesuai dengan KPI (*Key Performance Indicator*) yang sudah ditentukan di awal. Trafik data yang mengalir pada jaringan LAN akan bertumpu pada jalur (*link*) yang menghubungkan antar perangkat switch. Semakin banyak jumlah komputer yang dihubungkan dan semakin besar data yang dipertukarkan antar komputer akan menyebabkan masalah *bottleneck* pada jalur *trunking* apabila tidak terdapat perencanaan jaringan LAN dengan baik. Permasalahan *bottleneck* terjadi apabila *bandwidth* yang disediakan pada jalur *trunking* lebih kecil atau sama dengan jalur akses yang menghubungkan antara perangkat switch dengan komputer. Peristiwa *bottleneck* merupakan sebuah keadaan dimana kecepatan jumlah data yang masuk perangkat switch lebih besar dari kecepatan proses pengiriman data pada jalur *trunking*. Penurunan performansi jaringan dari sisi *delay* pengiriman data menjadi semakin besar. Pengaturan trafik data juga bisa dilakukan pada jaringan *mobile* dari perangkat yang selalu bergerak. Jaringan VANET (*Vehicular Ad hoc Networks*) merupakan salah satu contoh jaringan *mobile*. Kecenderungan perangkat yang selalu bergerak cepat dan sistem komunikasi menggunakan media *wireless* menjadikan tantangan tersendiri untuk membuat algoritma pengaturan trafik data pada jaringan VANET. Salah satu algoritma yang digunakan adalah CRONOS (Costa, dkk, 2018) yang dapat menurunkan *overhead* trafik data dan cakupan jaringan yang luas. Komunikasi V2V (*Vehicle to Vehicle*) dengan kecepatan kendaraan tinggi dan trafik lalu lintas padat juga akan mengakibatkan trafik data yang dibangkitkan antar kendaraan (*vehicle*) menjadi macet (*congestion*) karena *overhead* trafik data pada jaringan. Penggunaan protokol *DisTraC* juga bisa digunakan untuk mengurangi *overhead* trafik data (Sousa, dkk, 2020). Pengaturan *overhead* trafik data juga dilakukan pada jaringan SDN (*Software Defined Networks*). Penggunaan algoritma *Flowsense* dan *Payless* (Ren & Tang, 2020) bisa digunakan untuk memperkirakan kondisi trafik data pada jaringan SDN, sehingga pengaturan jalur trafik data ke jalur dengan kondisi trafik yang lebih rendah juga dapat mengurangi *overhead* trafik data dalam jaringan SDN.

Performansi jaringan menjadi pertimbangan utama ketika akan membangun sebuah jaringan. Kelancaran dalam proses bisnis sebuah perusahaan juga tergantung dari performansi jaringan yang digunakan oleh perusahaan tersebut (Wibowo, 2016). Salah satu cara yang digunakan untuk meningkatkan performansi jaringan yaitu dengan memisahkan wilayah jaringan menjadi bagian yang lebih kecil (Aziz, 2018). Pembagian wilayah jaringan akan membatasi data yang pola pengiriman data tersebut dilakukan secara *broadcast*. Pengiriman data dengan pola pengiriman secara *broadcast* dengan jumlah yang besar dapat membebani jaringan sehingga performansi jaringan menjadi menurun. Jaringan LAN sendiri akan menggunakan protokol *Ethernet* sebagai protokol layer 2 (*data link*) yang digunakan untuk proses pertukaran data

antar komputer (**Spurgeon & Zimmerman, 2014**). Permasalahan berikutnya pada jaringan yang menggunakan protokol *Ethernet* adalah ketika kejadian tabrakan data. Akibat adanya tabrakan data tersebut, komputer dalam wilayah jaringan yang sama akan mengalami proses *jamming* (**Kurose & Ross, 2017**). Semua komputer tidak dapat mengirim ataupun menerima data yang dikirimkan oleh perangkat lain dalam jaringan yang sama. Waktu pengiriman data menjadi lebih lama dengan adanya tabrakan data dalam jaringan. Sinyal *jamming* akan diterima oleh komputer yang satu wilayah jaringan yang sama. Semua komputer yang menerima sinyal *jamming* tersebut tidak bisa melakukan proses pengiriman atau penerimaan data. Jadi ketika terjadi masalah tabrakan data, maka komputer dalam satu wilayah jaringan yang akan terpengaruh dari masalah tersebut.

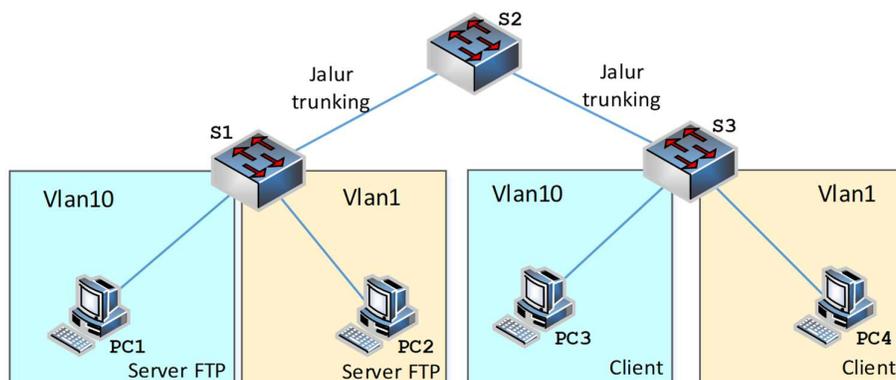
Pengiriman data secara *broadcast* biasanya dilakukan oleh pihak yang ingin mematikan sebuah *server* atau menurunkan performansi jaringan (**Virdaus, dkk, 2016**). Pengiriman data *broadcast* dengan jumlah yang sangat besar atau istilahnya dikenal dengan nama *broadcast storm* dapat terjadi karena ada percobaan serangan dari pihak ketiga (**Alimi & Mufutau, 2015**). Pola serangan dilakukan dengan cara membanjiri jaringan dengan data *broadcast*, sehingga *memory* perangkat tidak bisa lagi menampung data lain selain data *broadcast* tersebut. Pola serangan seperti ini dapat diatasi juga dengan menerapkan teknologi VLAN (*Virtual LAN*) (**Al-khaffaf, 2018**). Dengan membagi wilayah jaringan, maka wilayah edar dari data *broadcast* dapat dibatasi. Begitupula ketika terjadi tabrakan data, komputer yang berbeda wilayah jaringan tidak terpengaruh akibat adanya tabrakan data dari sebuah wilayah jaringan yang lain. Performansi dari sebuah jaringan juga dapat ditingkatkan dengan cara mendeteksi kondisi jaringan secepat mungkin. Deteksi awal akan adanya perubahan jaringan atau kegagalan saluran dapat juga meningkatkan performansi jaringan (**Kodama, dkk, 2016**). Pembagian wilayah *broadcast* selain dapat dilakukan dengan menggunakan perangkat yang bekerja di *layer 3* (*network*) seperti router juga bisa dilakukan dengan menggunakan perangkat yang bekerja di *layer 2* (*switch*). Semakin tinggi posisi *layer* menjadikan waktu proses enkapsulasi dan dekapsulasi perangkat menjadi semakin lama. Perangkat router akan membutuhkan waktu proses perangkat yang lebih lama ketika data diterima oleh *interface* router dibandingkan dengan perangkat switch. Sebagai solusi dari permasalahan ini, teknologi VLAN dapat dijadikan solusi dalam memecah wilayah *broadcast* dalam sebuah jaringan LAN selain menggunakan perangkat router. Perangkat yang bisa diimplementasikan teknologi VLAN adalah switch. Teknologi VLAN menggunakan konsep membagi jaringan LAN secara *virtual*. Jaringan LAN identik dengan jaringan yang menggunakan switch sebagai perangkat penghubung. Teknologi VLAN dapat diterapkan pada perangkat switch bukan router, sehingga jaringan yang menggunakan perangkat switch dapat diimplementasikan penggunaan teknologi VLAN. Penggunaan teknologi VLAN memberikan beberapa keuntungan atau kelebihan diantaranya dapat mengefisienkan penggunaan perangkat penghubung, konfigurasi lebih mudah dan menyediakan fasilitas keamanan data (**Saputra & Suryawan, 2018**).

Jaringan LAN yang menghubungkan beberapa perangkat switch menggunakan konsep jalur *trunking* dalam mempertukarkan data antar perangkat switch. Jalur *trunking* sendiri adalah jalur penghubung antar perangkat switch. Data yang dipertukarkan pada jaringan LAN akan menggunakan jalur *trunking*. Penggunaan jalur *trunking* untuk mempertukarkan data akan menimbulkan kekurangan disisi *overhead traffic*. Istilah ini menggambarkan sebuah kondisi dimana terdapat penambahan *header* pada sebuah data yang berisi informasi tambahan yaitu asal wilayah VLAN dari data yang akan dilewatkan pada jalur *trunking*. Konsep *header* dalam sebuah data merupakan informasi tambahan selain data yang berisi sebuah informasi. Proses pertukaran data dalam sebuah jaringan tidak hanya sekedar mempertukarkan data melainkan juga mengikutkan informasi tambahan berupa *header* ke dalam data tersebut. Dengan kata lain, *header* dalam sebuah data merupakan data tambahan yang dapat membebani jaringan

itu sendiri apabila ukuran dari *header* cukup besar dan dikirimkan dengan jumlah yang besar. Konsep jaringan LAN yang menerapkan teknologi VLAN akan memanfaatkan jalur *trunking* dalam mempertukarkan data antar wilayah VLAN yang berbeda. Identitas dari sebuah data yang melewati jalur *trunking* diberikan dengan menerapkan konsep VLAN ID atau identitas asal dari sebuah data yang melewati jalur *trunking* (Empson, 2016). Ketika data melewati jalur *trunking*, informasi VLAN ID dalam sebuah *header* ditambahkan ke dalam data yang melewati jalur *trunking*. Proses ini akan menambahkan *overhead* trafik data yang melewati jalur *trunking* itu sendiri. Semakin besar jumlah trafik data yang melewati jalur *trunking*, semakin besar pula ukuran *overhead* trafik yang membebani jaringan LAN, terlebih dengan semakin banyak data *broadcast* yang dilewatkan pada jalur *trunking*. Data yang sifat pengirimannya secara *broadcast* sendiri biasanya dibangkitkan disaat penggunaan protokol ARP (*Address Resolution Protocol*) atau DHCP (*Dynamic Host Configuration Protocol*). Paket ARP *Request* dalam mencari alamat MAC komputer lawan atau paket DHCP *Discover* akan dikirimkan secara *broadcast* (Thu, 2018). Perangkat switch produk dari Cisco memiliki fasilitas yang dinamakan sebagai *native* VLAN dalam mengurangi *overhead* trafik data yang melewati jalur *trunking*. Perangkat switch tidak akan menambahkan *header* pada sebuah data yang akan melewati jalur *trunking* jika diketahui asal wilayah VLAN dari data yang dilewatkan sama dengan informasi *native* VLAN yang digunakan oleh perangkat switch. Penelitian ini bertujuan untuk mengetahui performansi jaringan LAN yang menggunakan perangkat switch produk dari Cisco yang sudah menerapkan *native* VLAN. Penelitian menggunakan *software* emulasi GNS3 dalam membangun jaringan LAN yang menerapkan *native* VLAN. Jaringan LAN akan dibangun menggunakan *software* tersebut. Dalam menjalankan dan mengkonfigurasi perangkat switch dilakukan langsung pada *software* GNS3 dengan menjalankan GNS3 VM (*Virtual Machine*). Pengujian jaringan dilakukan dengan menggunakan ukuran data yang berbeda dimulai dari 10 MByte sampai 100 MByte. Perubahan ukuran data tersebut digunakan untuk mengukur prosentase beban jaringan atau *overhead* trafik data dan mengetahui performansi jaringan LAN dengan menggunakan parameter *delay*, *jitter*, dan *throughput* dengan mengimplementasikan fasilitas *native* VLAN pada jaringan LAN.

2. METODOLOGI PENELITIAN

Penelitian ini menggunakan perangkat lunak GNS3 dalam merancang dan membangun jaringan uji. Penelitian dilakukan dengan berbasiskan lab menggunakan perangkat lunak GNS3. Perangkat yang digunakan pada jaringan uji akan dihubungkan dan dikonfigurasi menggunakan *syntax* untuk perangkat produk dari Cisco. Perangkat lunak GNS3 digunakan sebagai emulator agar perangkat switch dan komputer yang digunakan pada jaringan uji mendekati karakteristik yang dimiliki oleh perangkat asli dalam jaringan. Penelitian menggunakan satu topologi jaringan yang sama seperti yang terlihat pada keterangan Gambar 1, dimana teknologi VLAN akan diaktifkan pada perangkat switch yang digunakan pada topologi jaringan uji. Skenario yang digunakan pada saat membangun jaringan uji adalah dengan menerapkan konsep komunikasi antar komputer antar dua wilayah VLAN yang berbeda. Pengujian dilakukan pada masing-masing wilayah VLAN. Pertukaran data antar komputer pada wilayah VLAN yang berbeda dilakukan dengan menggunakan ukuran data yang berbeda. Hal ini dilakukan untuk melihat dan membandingkan performansi jaringan VLAN ketika perangkat switch meneruskan data pada jalur *trunking* dengan menerapkan fasilitas *native* VLAN.



Gambar 1. Topologi jaringan uji yang menerapkan VLAN

Topologi jaringan yang digunakan untuk menguji penggunaan fasilitas *native* VLAN pada jaringan LAN terlihat pada keterangan Gambar 1. Empat komputer digunakan untuk menguji fasilitas *native* VLAN dengan cara saling mempertukarkan data dengan ukuran yang bervariasi dimulai dari ukuran 10 Mbyte sampai maksimal 100 Mbyte. Fasilitas *native* VLAN akan diaktifkan pada masing-masing switch, sehingga apabila ada data yang dilewatkan pada jalur *trunking*, maka akan terdapat proses penambahan *header* terkait asal wilayah VLAN dari data yang dilewatkan pada jalur *trunking* tersebut. Namun tidak semua data yang dilewatkan pada jalur *trunking* akan mendapatkan proses penambahan *header*. Pada contoh kasus penggunaan perangkat switch produk dari Cisco yang dinamakan sebagai *Catalyst*, data yang dilewatkan pada jalur *trunking* tidak mendapatkan proses penambahan *header* apabila asal wilayah VLAN dari data yang dilewatkan adalah sama dengan informasi ID dari *native* VLAN. Hal ini yang mengakibatkan terjadinya perbedaan *overhead* trafik data pada jalur *trunking* antara data yang menggunakan informasi VLAN-ID yang sama dan berbeda dengan *native* VLAN. Perangkat switch yang digunakan pada topologi jaringan uji menggunakan produk dari Cisco, sehingga ID awal yang digunakan oleh perangkat switch tersebut untuk informasi *native* VLAN adalah 1 (VLAN ID = 1).

Pengujian dilakukan menggunakan perangkat lunak GNS3 yang dapat digunakan untuk membangun jaringan dan menjalankan perangkat switch produk dari Cisco. Topologi jaringan menggunakan tiga perangkat switch yang saling terhubung dan antar perangkat switch akan membuat jalur *trunking* dengan kecepatan transfer maksimal sebesar 1 Gbps. Pengaturan kecepatan transfer data maksimal dilakukan pada perangkat switch dengan mempertimbangkan jalur *trunking* sebagai jalur utama dalam melewatkan trafik data antar wilayah VLAN yang berbeda. Perangkat lunak GNS3 sebagai perangkat lunak utama dalam membangun topologi jaringan uji menggunakan bantuan perangkat lunak tambahan yaitu *VMWare Workstation* sebagai perangkat lunak dalam membuat mesin atau komputer *virtual*. Konsep virtualisasi jaringan digunakan pada topologi jaringan uji dengan memanfaatkan perangkat lunak *VMWare Workstation* tersebut. Penggunaan konsep virtualisasi dikarenakan pengujian jaringan menggunakan perangkat switch yang bisa dikonfigurasi atau istilah lainnya adalah penggunaan perangkat switch yang *manageable* tidak bisa langsung menggunakan perangkat lunak GNS3. Diperlukan penggunaan sistem operasi perangkat switch seperti layaknya perangkat switch asli yang digunakan pada perangkat lunak GNS3, sehingga nantinya perangkat switch yang terpasang pada perangkat lunak GNS3 akan dihubungkan dengan sebuah komputer virtual tersendiri menggunakan *VMWare Workstation* yang digunakan untuk menjalankan sistem operasi perangkat switch. Pada pengujian jaringan, sistem operasi yang digunakan adalah Cisco IOSvL2.

```
switch(config)#interface <type interface>
switch(config-if)#switchport mode trunk
```

Gambar 2. Aktivasi jalur *trunking* pada perangkat switch

Pada awalnya, semua port switch yang digunakan pada topologi jaringan uji seperti yang terlihat pada Gambar 1 masih menggunakan mode *access*. Mode port switch ini hanya diperuntukkan jika *port* switch dihubungkan langsung dengan perangkat komputer. Ketika port switch dihubungkan dengan perangkat switch lain yang nantinya akan difungsikan sebagai jalur *trunking*, maka diperlukan mengubah mode *port* pada perangkat switch tersebut menjadi mode *trunk*. Keterangan Gambar 2 merupakan konfigurasi yang digunakan untuk mengubah dari mode *access* menjadi mode *trunk*. Ketika mode *port* diubah menjadi mode *trunk*, maka secara otomatis jalur *trunking* yang menghubungkan antar perangkat switch dan fasilitas *native* VLAN akan aktif pada perangkat switch tersebut. Pada keterangan Gambar 1, jalur *trunking* digunakan untuk melewati trafik data gabungan dari wilayah VLAN 1 dan VLAN 10. Perangkat switch akan membedakan trafik data yang berasal dari wilayah VLAN 1 dan VLAN 10. Karena fasilitas *native* VLAN aktif pada perangkat switch, maka ketika data dari wilayah VLAN 1 melewati jalur *trunking* dan menggunakan VLAN ID yang sama dengan *native* VLAN yaitu VLAN 1, maka tidak terdapat proses penambahan *header* pada data tersebut. Berbeda ketika data dari wilayah VLAN 10 melewati jalur *trunking*. Perangkat switch akan menambahkan informasi VLAN ID ke dalam *header* dari data yang dilewatkan pada jalur *trunking*. Proses penambahan *header* dilakukan ketika data masuk dan melewati jalur *trunking*. Perbedaan proses yang dilakukan oleh perangkat switch tersebut menjadikan alasan adanya proses analisa performansi jaringan LAN dengan mempertimbangkan pengaruh penggunaan fasilitas *native* VLAN.

3. HASIL DAN PEMBAHASAN

Pengujian jaringan LAN yang menerapkan fasilitas *native* VLAN dilakukan dengan tiga buah switch yang membentuk dua jalur *trunking*. Komunikasi antar komputer yang terletak pada dua wilayah VLAN yang berbeda akan menggunakan jalur *trunking* tersebut. Komunikasi antar komputer pada jaringan LAN akan menggunakan konsep *client-server*. Pada masing-masing wilayah VLAN terdapat satu komputer *server* yang digunakan untuk menyimpan data dengan ukuran tertentu, kemudian data tersebut akan diakses oleh komputer *client*. Skenario pengujian yang digunakan pada topologi jaringan terlihat pada Tabel 1.

Tabel 1. Parameter Simulasi

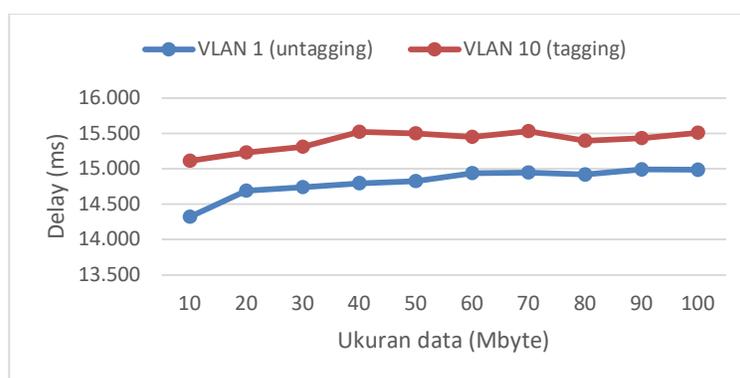
Parameter Simulasi	Nilai
Ukuran data minimal	10 Mbyte
Ukuran data maksimal	100 Mbyte
Jumlah VLAN-ID	2
<i>Native</i> VLAN	VLAN 1
Kecepatan koneksi jalur <i>trunk</i>	1 Gbps
Kecepatan koneksi jalur akses	100 Mbps
Jumlah PC <i>server</i>	2
Jumlah PC <i>client</i>	2

Konfigurasi VLAN dilakukan pada masing-masing switch baik itu pada switch S1, S2, dan S3. VLAN dengan ID 10 (VLAN 10) dibuat pada masing-masing switch. VLAN dengan ID 1 (VLAN

1) tidak perlu dibuat karena VLAN tersebut secara otomatis ada pada sebuah perangkat switch produk dari Cisco. Antar VLAN ID yang berbeda akan menggunakan jalur trunking dalam mempertukarkan data antara komputer *client* dan *server*. Parameter yang digunakan dalam proses pengujian jaringan LAN terdapat pada keterangan Tabel 1. Data dipertukarkan antara komputer *client* dan *server* dengan ukuran yang bervariasi, dimulai dari ukuran 10 Mbyte sampai 100 Mbyte dengan interval antar ukuran data sebesar 10 Mbyte. PC1 dan PC2 diposisikan sebagai *server* FTP yang digunakan untuk menyimpan *file* data dengan ukuran yang berbeda. *File* tersebut akan diakses oleh komputer *client* yaitu PC3 dan PC4 yang terletak pada wilayah VLAN yang berbeda. Pengujian dilakukan hanya antar komputer dengan wilayah VLAN yang sama, misalnya PC3 yang berada pada wilayah VLAN 10 sebagai *client* mengakses *file* data ke PC1 sebagai *server* FTP yang sama-sama berada pada wilayah VLAN yang sama. Begitupula pada proses komunikasi antara PC4 sebagai *client* mengakses *file* data ke PC2 sebagai *server* FTP yang berada pada wilayah VLAN 1. Perangkat switch yang digunakan adalah produk dari Cisco yang menggunakan informasi awal *native* VLAN yaitu VLAN 1. *Bandwidth* yang digunakan pada jalur *trunking* yaitu sebesar 1 Gbps. Pengamatan performansi jaringan LAN dilakukan dengan membedakan trafik data antara wilayah VLAN 10 dan VLAN 1, karena perangkat switch dengan informasi awal *native* VLAN yaitu VLAN 1 akan membedakan dalam proses enkapsulasi trafik data pada jalur *trunking* antara data yang berasal dari wilayah VLAN 10 dan VLAN 1.

3.1 Perhitungan Nilai *Delay*

Pengamatan performansi jaringan LAN salah satunya adalah dengan menggunakan parameter *delay*. Proses pengamatan dilakukan pada masing-masing komputer *client* yaitu PC3 dan PC4. Perangkat lunak *wireshark* diaktifkan pada kedua komputer tersebut, kemudian proses komunikasi antara PC3 dan PC4 sebagai komputer *client* ke masing-masing komputer *server* sesuai dengan wilayah VLAN dari masing-masing komputer *client* dilakukan untuk mengamati besaran nilai *delay*. Waktu yang dibutuhkan oleh data dari komputer *server* sampai ke komputer *client* dihitung dengan menggunakan bantuan perangkat lunak *wireshark*. Proses pengambilan data berupa nilai *delay* dilakukan pada sisi komputer *client* pada wilayah VLAN yang berbeda. Proses komunikasi antara PC1 dan PC3 yang terletak pada wilayah VLAN 10 dibuat terpisah dengan proses komunikasi antara PC2 dan PC4 karena terletak pada wilayah VLAN yang berbeda yaitu VLAN 1. Pada saat pengujian jaringan LAN, pengamatan performansi tidak dilakukan pada komunikasi antar komputer yang terletak pada wilayah VLAN yang berbeda. Hasil pengukuran nilai *delay* antara komunikasi komputer *client* dan *server* pada wilayah VLAN yang berbeda terdapat pada keterangan Gambar 3. Proses pengambilan nilai *delay* dilakukan dengan mempertukarkan data dengan dimulai dari ukuran 10 Mbyte sampai maksimal 100 Mbyte.

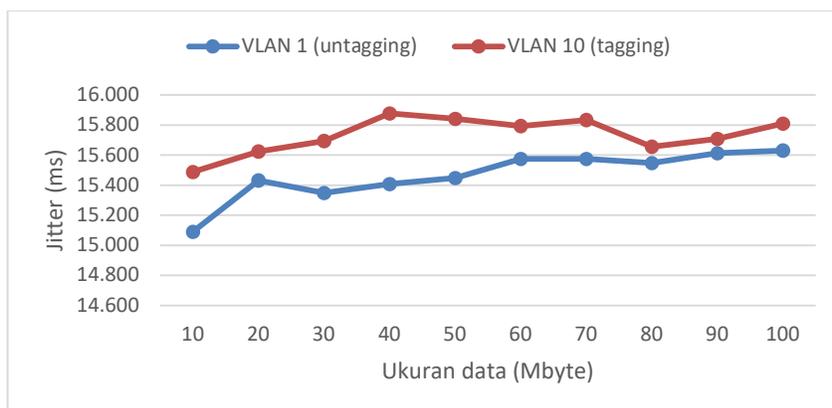


Gambar 3. Nilai *delay* antara VLAN 1 dan VLAN 10

Perangkat switch akan melakukan proses enkapsulasi ketika data tersebut akan melewati jalur *trunking*. Data akan ditambahkan *header* yang berisi informasi wilayah VLAN asal dari data yang akan dikirimkan. Proses ini dinamakan dengan istilah *tagging*. Kebalikan dari proses tersebut adalah *untagging*. Data yang dilewatkan jalur *trunking* tidak mendapatkan tambahan *header* asalkan informasi ID dari wilayah VLAN asal data yang dilewatkan jalur *trunking* sama dengan ID dari *native* VLAN. Hal ini yang dilakukan oleh switch S1 ketika ada data yang dikirimkan oleh PC2 sebagai server VTP pada wilayah VLAN 1. Switch S1 tidak menambahkan header ke dalam data dari PC2 yang dikirimkan ke PC4 sebagai *client* pada wilayah VLAN yang sama yaitu VLAN 1. Berbeda ketika switch menerima data dari PC1 sebagai *server* VTP untuk PC3. Wilayah VLAN yang digunakan oleh kedua PC tersebut adalah VLAN 10, sehingga data yang datangnya dari PC1 akan ditambahkan *header* berisi informasi VLAN sebesar 4 byte ke dalam data yang dikirimkan oleh PC1. Proses penambahan *header* (*tagging*) tersebut menghasilkan nilai *delay* yang lebih besar dibandingkan data yang tidak mendapatkan *header* (*untagging*). Hal ini terlihat dari perbandingan hasil nilai *delay* antara proses pertukaran data antar PC yang terletak pada wilayah VLAN 1 dan VLAN 10 pada Gambar 3. Pertambahan ukuran data yang dipertukarkan antara komputer *client* dan *server* menjadikan nilai *delay* yang dihasilkan semakin besar. Besaran nilai *delay* untuk pertukaran data sebesar 50 Mbyte pada wilayah VLAN 1 menghasilkan nilai *delay* sebesar 14,824 ms. Nilai *delay* tersebut menjadi lebih besar ketika digunakan antara PC1 sebagai *server* VTP dengan PC3 sebagai *client* yang terletak pada wilayah VLAN 10 yaitu sebesar 15,523 ms.

3.2 Perhitungan Nilai *Jitter*

Data yang dikirimkan antar komputer yang terletak pada wilayah VLAN yang berbeda akan melewati jalur *trunking*. Semakin besar ukuran data, semakin banyak jumlah pecahan data yang harus yang dipertukarkan antara komputer *server* dan *client*. Perbedaan interval waktu pengiriman data yang dikirimkan antara komputer *client* dan *server* yang dikenal dengan istilah *jitter* diukur dan ditampilkan pada keterangan Gambar 4.

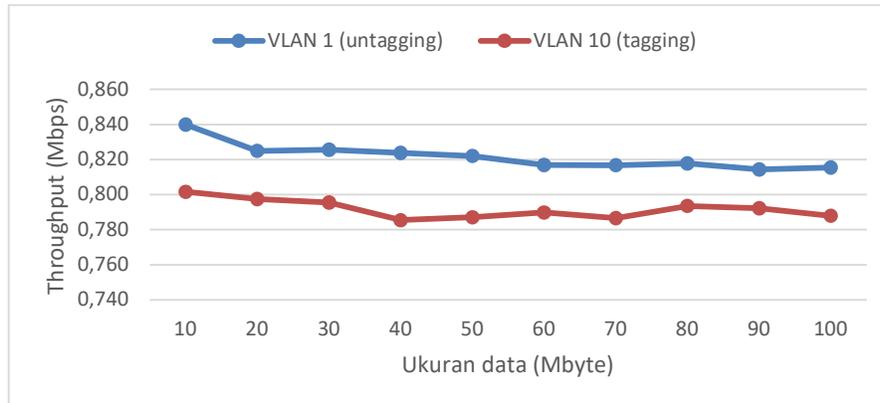


Gambar 4. Nilai *jitter* antara VLAN 1 dan VLAN 10

Hasil keluaran pengujian parameter *jitter* memiliki tren naik dengan semakin besar ukuran data yang dipertukarkan antara *server* VTP dan *client* pada masing-masing wilayah VLAN. Nilai yang dihasilkan dari parameter *jitter* memiliki tren yang sama dengan nilai parameter *delay*. Penggunaan fasilitas *native* VLAN pada saat proses pengiriman data untuk komputer yang berada pada wilayah VLAN 1 memberikan nilai *jitter* yang lebih kecil dibandingkan dengan wilayah VLAN 10. Perbaikan dari sisi parameter nilai *jitter* ketika menggunakan *native* VLAN berakibat pada kecepatan dalam proses pertukaran data antara komputer *client* dan *server*.

3.3 Perhitungan Nilai *Throughput*

Pengukuran nilai *throughput* dilakukan pada sisi komputer *client* dengan menggunakan perangkat lunak *wireshark*. Pada saat *file* data diambil dari komputer *server*, perangkat lunak *wireshark* diaktifkan pada komputer *client* untuk mengetahui besaran nilai *throughput* yang digunakan oleh komputer *client*. *Bandwidth* yang terpasang pada *interface* komputer *client* adalah sebesar 100 Mbps dengan tipe *interface* FastEthernet. Pengukuran nilai *throughput* dilakukan pada proses pertukaran data antar wilayah VLAN yang berbeda.



Gambar 5. Hasil pengukuran nilai throughput antara VLAN 1 dan VLAN 10

Besaran nilai *throughput* akan lebih kecil atau sama dengan *bandwidth* yang terpasang pada *interface* yang digunakan untuk menerima dan mengirimkan data. Keterangan Gambar 5 memperlihatkan hasil pengukuran parameter *throughput* pada masing-masing komputer *client* menghasilkan nilai yang lebih kecil dari *bandwidth* yang disediakan oleh *interface* komputer *client*. Nilai *throughput* tertinggi yang dihasilkan ketika ukuran data yang dipertukarkan sebesar 10 Mbyte antara PC1 dan PC3 yang terletak pada wilayah VLAN 10 yaitu sebesar 839,86 Kbps (0,84 Mbps). Ketika data dengan ukuran yang sama dipertukarkan antara komputer *server* dan *client* tidak mendapatkan proses penambahan *header* pada perangkat switch yaitu antara PC2 dan PC4 yang terletak pada wilayah VLAN 1 menghasilkan nilai *throughput* yang lebih kecil yaitu sebesar 801,571 Kbps (0,80 Mbps). Penggunaan *native* VLAN pada saat proses pertukaran data antar komputer yang terletak pada wilayah VLAN 1 menghasilkan nilai *throughput* yang lebih kecil dibandingkan VLAN 10 dikarenakan dengan menambah *header* ke dalam data ketika data dilewatkan jalur *trunking* akan menambah ukuran data yang dilewatkan. Penambahan *header* akan menambah *overhead* dari data yang dilewatkan pada jalur *trunking* juga menambah besaran nilai *throughput*. Penurunan hasil nilai *throughput* terjadi dengan menambah ukuran data yang dipertukarkan antara komputer *client* dan *server*.

4. KESIMPULAN

Pengujian jaringan LAN yang menggunakan teknologi VLAN dan fasilitas *native* VLAN dapat menurunkan *overhead* trafik data yang melewati jalur *trunking*. *Native* VLAN dapat mengurangi ukuran dari setiap data yang dilewatkan yaitu sebesar 4 Byte. Pengurangan ukuran data yang dilewatkan dalam jalur *trunking* dapat meningkatkan performansi jaringan LAN itu sendiri baik dari sisi *delay* atau *throughput*. Pada pengujian pertukaran data sebesar 50 Mbyte antara PC2 dan PC4 yang terletak pada wilayah VLAN 1 memberikan perbaikan nilai *delay* sebesar 4,72% dibandingkan dengan proses pertukaran data antara PC1 dan PC3 yang terletak pada wilayah VLAN 10. Perangkat switch yang diaktifkan fasilitas *native* VLAN akan

menambahkan *header* berisi informasi asal wilayah VLAN dari data yang dikirimkan melewati jalur *trunking* hanya ketika data yang dilewatkan bukan berasal dari wilayah VLAN 1. Performansi jaringan LAN juga memberikan perbaikan sebesar 4,77% dilihat dari sisi parameter *throughput* ketika dibandingkan antara proses pertukaran data dari wilayah VLAN 1 dengan VLAN 10. Pembebanan trafik data pada jaringan LAN akibat adanya proses enkapsulasi tambahan pada jalur *trunking* menggunakan pendekatan parameter *throughput*, karena parameter tersebut berkaitan dengan besaran penggunaan data oleh *interface* komputer. Pengujian lanjutan dapat dilakukan dengan memberikan trafik data dengan ukuran yang lebih besar, misalnya untuk aplikasi *video conference* atau ketika terjadi pengaruh serangan dari pihak luar dengan cara mengirimkan data *broadcast* ke dalam jaringan LAN.

DAFTAR RUJUKAN

- Al-khaffaf, D. A. J. (2018). Improving LAN Performance Based on IEEE802 . 1Q VLAN Switching Techniques. *Journal of University of Babylon*, (1), 286–297.
- Alimi, I. A., & Mufutau, A. O. (2015). Enhancement of Network Performance of an Enterprises Network with VLAN. *American Journal of Mobile Systems*, 1(2), 82–93.
- Aziz, D. A. (2018). The Importance of VLANs and Trunk Links in Network Communication Areas. *International Journal of Scientific & Engineering Research*, 9(9), 10–15.
- Costa, J., Rosario, D., De Souza, A. M., Villas, L. A., & Cerqueira, E. (2018). Data Dissemination Based on Complex Networks' Metrics for Distributed Traffic Management Systems. *Proceedings - IEEE Symposium on Computers and Communications, 2018-June*, (pp. 1062–1067). <https://doi.org/10.1109/ISCC.2018.8538601>
- Sousa, R. S., Boukerche, A., & Loureiro, A. A. F. (2020). A distributed and low-overhead traffic congestion control protocol for vehicular ad hoc networks. *Computer Communications*, 159, 258–270. <https://doi.org/10.1016/j.comcom.2020.05.032>
- Empson, S. (2016). *CCNA Routing and Switching Portable Command Guide*. Indianapolis: Cisco Press.
- Khalaf, A. A. M., Mokadem, M. S. A., & Ahmad, K. A. (2015). Performance of LAN under different ethernet wiring standard and different frame size. *International Conference on Advanced Communication Technology, ICACT, 2015-Augus*, (pp. 638–647). <https://doi.org/10.1109/ICACT.2015.7224875>
- Kodama, S., Nakagawa, R., Tanouchi, T., & Kameyama, S. (2016). Management system by using embedded packet for hierarchical local area network. *2016 IEEE 7th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference, UEMCON 2016*, (pp. 5–8). <https://doi.org/10.1109/UEMCON.2016.7777868>
- Kurose, J. F., & Ross, K. W. (2017). *Computer Networking (seventh edition)*. Boston: Pearson Education.

- Nugroho, K., & Fallah, M. S. (2018). Implementasi Load Balancing menggunakan Teknologi EtherChannel pada Jaringan LAN. *ELKOMIKA: Jurnal Teknik Energi Elektrik, Teknik Telekomunikasi, & Teknik Elektronika*, 6(3), 420. <https://doi.org/10.26760/elkomika.v6i3.420>
- Ren, S., & Tang, G. (2020). A reactive traffic flow estimation in software defined networks. *2020 5th International Conference on Computer and Communication Systems, ICCCS 2020*, (pp. 585–588). <https://doi.org/10.1109/ICCCS49078.2020.9118430>
- Saputra, W., & Suryawan, F. (2018). Implementasi VLAN dan Spanning Tree Protocol Menggunakan GNS 3 dan Pengujian Sistem Keamanannya. *Khazanah Informatika: Jurnal Ilmu Komputer Dan Informatika*, 3(2), 64. <https://doi.org/10.23917/khif.v3i2.5311>
- Spurgeon, C., & Zimmerman, J. (2014). *Ethernet: The Definitive Guide, 2nd Edition: Designing and Managing Local Area Networks*. USA: O'Reilly Media.
- Thu, K. A. (2018). *Understanding, Design and Configuration of Virtual Local Area Networks*. 7(10), 746–751.
- Virdaus, I. K., Kang, M., Shin, S., & Kwon, G. R. (2016). A simulation study: Is the broadcast storming really harmful for emergency delivery in VANETs? *International Conference on Advanced Technologies for Communications, 2016-Janua*, 666–670. <https://doi.org/10.1109/ATC.2015.7388415>
- Wibowo, E. (2016). Pemanfaatan Teknologi E-Commerce Dalam Proses Bisnis. *EQUILIBRIA*, 1(1).