ANALISIS IMPLEMENTASI COS UNTUK MENGATASI MASALAH *DELAY*, *JITTER*, *PACKETLOSS* MENGGUNAKAN *CBWFQ PCQ*

COS IMPLEMENTATION ANALYSIS FOR SOLVING DELAY, JITTER, AND PACKET LOSS PROBLEMS USING CBWFQ PCQ

Muhamad Hikam¹⁾, Raka Yusuf²⁾

^{1), 2)} Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana
Jl Raya Meruya, RT.4/RW.1, Kec. Kembangan, Jakarta Barat, 11650
Email : 41517110066@student.mercubuana.ac.id¹, raka@mercubuana.ac.id²

Abstrak

Menganalisis penyebab dari *delay time*, *jitter ping* dan *packet loss* yang tinggi dalam sebuah *network layer 2 switching* yang dapat mempengaruhi performansi layanan yang diberikan, kemudian mengimplementasikan *Class Of Service (COS)* yang tepat sehingga dapat memberikan layanan yang lebih optimal. Tujuan dari penelitian ini adalah untuk mengatasi *delay time*, *jitter ping* dan *packet loss* yang terjadi sehingga penggunaan *bandwidth* lebih maksimal dan efisien. Metode penelitian yang digunakan adalah dengan membandingkan dua algoritma *queueing* yaitu *CBWFQ (Class Based Weighted Fair Queueing)* yang diterapkan pada sisi *backhaul layer 2 switching* dan *PCQ (Per Connection Queue)* yang diterapkan pada sisi *end to end*. Kesimpulan yang dapat diambil dari penelitian ini adalah pada umumnya dalam koneksi *layer 2 switching* semua *packet data* baik itu yang dikirim maupun *packet data* yang diterima tidak ada *priority* didalamnya, sehingga setiap *packet data* yang dikirimkan pertama kali maupun dikirim secara bersamaan tidak akan ada pengaturan *bandwidth priority* di dalamnya yang dapat menyebabkan *congestion traffic*. Fungsi (*Class Of Service*) *COS* di dalam *network layer 2 switching* ini memberikan *class priority* sesuai dengan kebutuhan customer, layanan apa yang ingin diprioritaskan. Sehingga manajemen *bandwidth* dapat lebih teratur sesuai dengan kebutuhan customer.

Kata kunci: *Queueing, Delay, Jitter, Packet Loss*

Abstract

Analyze suspect of *delay time*, *ping jitter* and high *packet loss* in a *layer 2 switching network* which can affect the given performance, then implement the appropriate *Class of Service (COS)* so that it can provide more optimal service. The purpose of this research is to overcome the *delay time*, *ping jitter* and *packet loss* that occurs so that *bandwidth usage* is maximized and efficient. The research method used is to compare two *queueing algorithms*, *CBWFQ (Class Based Weighted Fair Queueing)* which is applied to the *backhaul layer 2 switching* and *PCQ (Per Connection Queue)* side which is applied to the *end-to-end side*. The conclusion that can be drawn from this research is that in general, in a *layer 2 switching connection*, all *packet data*, whether sent or received, has no *priority*, so that the *packet data* sent for the first time or sent simultaneously will not have any *priority bandwidth settings* inside which can cause *traffic jams*. The *COS (Class Of Service)* function in the *layer 2 switching network* provides *class priority* according to customer needs, what services do you want to prioritize. So that *bandwidth management* can be more organized according to customer needs.

Keywords : *Queueing, Delay, Jitter, Packet Loss*

1. PENDAHULUAN

Kebutuhan jaringan internet yang dapat bekerja selama 24 jam 365 hari adalah sebuah keharusan pada era kemajuan dan perkembangan teknologi yang pesat seperti sekarang ini. Akan tetapi selama ini kebutuhan akan jaringan internet yang stabil dan baik tersebut tidak didukung dengan *management bandwidth* yang baik[1]. Akibat dari *management bandwidth* yang tidak baik dapat menimbulkan masalah seperti *packet loss* yang akan menyebabkan *jitter ping* dan *delay time* yang tinggi sehingga layanan yang diberikan tidak berjalan maksimal. Lebih buruknya lagi *packet loss* tersebut juga dapat berdampak *downtime* terhadap layanan.

Delay time, *jitter ping* serta *packet loss* yang tinggi tentunya dapat mempengaruhi performansi layanan yang diberikan. Dampaknya terhadap *user* seperti lambatnya dalam mengakses internet serta *delay time* saat melakukan komunikasi suara dan video yang membutuhkan *bandwidth realtime*. Masalah yang cukup fatal kedepannya jika tidak ada perbaikan *packet loss* itu sendiri dapat menyebabkan *congestion* pada *traffic*[2].

Untuk mengatasi masalah yang dihadapi diperlukan sebuah standar layanan yang diterapkan dalam *network layer 2 switching* dengan cara melakukan penerapan standar *COS (Class Of Service)*. Standar tersebut ditentukan dengan menganalisa *user experience* karena tentunya setiap *user* pasti mempunyai karakteristik penggunaan kebutuhan internet yang berbeda. Penulis melakukan analisa & implementasi *COS* tersebut di PT. Iforte Global Internet tempat penulis bekerja dengan membandingkan metode *CBWFQ (Class Based Weight Fair Queue)* dan *PCQ (Peer Connection Queue)*[3].

2. DASAR TEORI

2.1. Class Of Service (COS)

COS (Class Of Service) adalah parameter standar layanan yang digunakan dalam protokol data dan *voice* untuk membedakan jenis *payloads* yang terdapat dalam paket yang dikirim. Tujuan dari diferensiasi tersebut umumnya terkait dengan penetapan prioritas ke *payloads* data atau tingkatan akses. *COS* hanya beroperasi pada 802.1Q *VLAN Ethernet* pada lapisan *data link (layer 2)*, sedangkan mekanisme *QOS* lainnya (seperti *DiffServ*, juga dikenal sebagai *DSCP*) beroperasi pada lapisan jaringan *IP (layer 3)* atau menggunakan sistem penandaan *QOS* lokal yang tidak mengubah paket sebenarnya, seperti "*QOS-Group*" Cisco[4].

2.2. Delay, Jitter & Packet Loss

Delay adalah waktu tunda sebuah proses pengiriman data dari satu titik sumber ke titik tujuan. Seperti di jalan raya jika dalam perjalanan ada hal-hal yang terjadi tidak terduga maka akan membuat proses perjalanan kita terganggu alias tidak sesuai target. Hal ini mungkin dalam perjalanan seringnya menjumpai lampu merah, maka perjalanan akan tertunda dari target yang diinginkan. Dalam jaringan internet sering terjadi *delay* pada saat media transmisi mengalami kestabilan[5].

Jitter adalah kumpulan dari semua *delay* yang terjadi selama proses data dikirimkan sampai dengan data diterima, *jitter* yang terjadi mendekati nol maka kecepatan jaringan sangat cepat. Namun sebaliknya jika tidak mendekati nol maka kecepatannya jelek dan akan terjadi kehilangan data dalam proses pengirimannya.

Packet Loss adalah gagalnya proses transmisi data kepada alamat tujuan yang menyebabkan hilangnya beberapa data dalam proses pengiriman.

2.3. Network Layer 2 Switching

Layer 2 atau *layer data link* adalah bagaimana jaringan terhubung melalui *data link*. Di *layer* ini ada perangkat-perangkat yang terhubung secara *link*, perangkat *layer 2* contohnya *switch* dan *hub*. Perangkat di *layer 2* ini masing-masing punya identitas yang disebut *MAC address*. Di *layer 2*, ada protokol *Ethernet*. Di protokol ini, paket *Ethernet* mempunyai alamat asal dan alamat tujuan. Alamat tersebut disebut alamat *MAC (Media Access Control)* atau *MAC Address*. Format alamat *MAC* adalah 48 bit bilangan heksadesimal, berikut merupakan contoh dari *MAC address* 80:C1:6E:59:2E:43. Seperti dijelaskan sebelumnya, setiap perangkat *layer 2* punya *MAC Address* yang unik[6].

3. METODOLOGI PENELITIAN

Metodologi penelitian yang digunakan untuk menganalisis dan implementasi *COS (Class Of Service)* pada *network layer 2 switching* adalah metodologi *NDLC (Network Development Life Cycle)*[7].

Gambar 1. Metodologi Penelitian

Pada gambar 1 metode NDLC dimulai dari tahap mengumpulkan data seperti *trend* grafik PRTG (*Paessler Router Traffic Grapher*), *log router user* dan *log modem*. Tahapan selanjutnya jika data sudah cukup dalam proses analisis fokus berdasarkan kebutuhan *user experience*. Masuk ke dalam tahap simulasi konfigurasi COS (*Class Of Service*) dengan cara melakukan *sampling* konfigurasi beberapa *site*, apabila simulasi konfigurasi COS berhasil maka tahapan selanjutnya adalah implementasi konfigurasi COS secara keseluruhan dan melakukan *monitoring* kembali setelah dilakukan implementasi.

4. PENGUJIAN DAN PEMBAHASAN

4.1. Analisis Berdasarkan Kebutuhan User

Tahapan ini menurut penulis merupakan tahapan yang sangat penting karena akan menentukan seperti apa *trend* penggunaan *bandwidth* yang akan dihasilkan. Akan tetapi, tahapan ini justru sering dikesampingkan bahkan dilupakan oleh para *network administrator*. Kebutuhan *user* dapat dilihat pada Tabel 1 dan Tabel 2. Pada Tabel 1 *list COS Priority* yang sudah ditetapkan ini disesuaikan dengan kebutuhan seperti *VLAN label* yang ingin diprioritaskan *monitoring* untuk *SLA (Service Level Agreement)*, *link 2G* dan prioritas selanjutnya. Pada Tabel 2 menginformasikan *VLAN number* yang sudah disesuaikan dengan kebutuhan *user*[8].

Tabel 1. Tabel COS Priority

<i>COS Priority</i>	<i>VLAN</i>	<i>Label</i>
1	<i>Monitoring SLA</i>	<i>Interactive</i>
2	<i>2G</i>	<i>Interactive</i>
3	<i>4G</i>	<i>Bulk</i>

Tabel 2. *VLAN Priority*

<i>OPSEL</i>	<i>VLAN</i>	<i>COS Priority Label</i>
<i>PRTG</i>	2861	<i>Interactive</i>
<i>OAM BTS</i>	1001 & 1011	<i>Interactive</i>
<i>2G</i>	201 & 211	<i>Interactive</i>
<i>4G</i>	401 & 411	<i>Bulk</i>

4.2. Analisis Berdasarkan Segmentasi Topologi Jaringan

Segmentasi jaringan adalah tindakan atau praktik memecah jaringan komputer menjadi *Subnetwork*, masing-masing menjadi segmen jaringan. Keuntungan dari pemisahan tersebut terutama untuk meningkatkan kinerja dan meningkatkan keamanan. Umumnya untuk dapat melakukan *troubleshooting* pada *network layer 2 switching* kita harus melakukan segmentasi masalah yang terjadi melalui topologi *network* yang sudah kita susun, baik itu dari sisi *backhaul* internet sampai dengan jaringan *end-to-end*. Contoh dapat dilihat pada gambar 2. Dari topologi kita dapat mengecek setiap *log* perangkat apakah terdapat *log error* yang terjadi[9].

Gambar 2. *Topologi Network Layer 2*

Contoh gambar 3, *MAC address flapping* pada router *ASR backhaul* ini mengindikasikan masalah yang terdapat pada *network layer 2* biasanya disebabkan karena adanya *duplicate MAC address* yang muncul dari sisi user. Untuk dapat mengetahui sumber dari *flapping* tersebut kita dapat melakukan *trace* dari setiap jalur *VLAN* mulai dari *backhaul* internet sampai dengan sisi *user*[10].

```

Oct 3 07:00:42.011: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:977b:450e in bridge domain 14 is flapping between port Gi0/0/0.Efp1508 and Gi0/0/0.Efp1421
Oct 3 07:00:42.473: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:977b:450e in bridge domain 14 is flapping between port Gi0/0/0.Efp1421 and Gi0/0/0.Efp1508
Oct 3 07:00:42.774: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:00:45.431: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:00:47.730: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:00:48.212: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:00:50.647: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:00:53.086: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:00:55.498: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:00:55.960: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:00:58.384: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:01:00.077: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:01:03.242: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:01:07.706: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:01:08.209: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:01:10.540: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:01:11.114: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:01:12.579: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:01:13.913: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
Oct 3 07:01:15.491: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/2.Efp4 and Gi0/0/0.Efp144
Oct 3 07:01:15.953: %IOSXE-3-PLATFORM: SIP0: cylon_mgr: MAC_FLAP: Host 0xd8c4:9773:4710 in bridge domain 14 is flapping between port Gi0/0/0.Efp1445 and Gi0/0/2.Efp144
ASR9025#
  
```

Gambar 3. MAC Address Flapping

Untuk memudahkan dalam mencari sumber permasalahan *MAC flapping* pada sisi *user*, kita dapat melakukan *sniffing packet* pada sisi *backhaul* sampai ke dalam *network user* dengan menggunakan *tools sniffing* yang tersedia. Jika *MAC address user* yang bermasalah sudah ditemukan maka dapat kita cek juga pada sisi *modem user*, seperti contoh pada gambar 4.

SRC MAC	DEST MAC	COS	PRI	Rule	PktCnt	Inactive Secs	Hash Idx	VLAN ID	VLAN PCP	PROTO	EVC ID	PkSent	PkDrop	PkQueued	ActvJoin	ActvLeave
28.7b:09:af:54:ef	00:00:0c:9f:19:fa	3	3	3	640015	0	292	2554	0	0x0800	19018920401366_EVC3(3)	640015	0	0	635777	635777
28.7b:09:af:54:ef	00:00:0c:9f:19:96	3	3	3	1123456	0	292	2454	0	0x0800	19018920401366_EVC3(5)	1122831	625	0	1096550	1096550
28.7b:09:af:54:ef	00:00:0c:9f:19:91	1	1	1	14	1	551	2705	0	0x0800	19018920401366_EVC2(2)	14	0	0	14	14
74.4d:28:b8:61:e4	6c:6c:d3:20:67:62	1	1	1	1	10	1009	2861	0	0x0806	19018920401366_EVC6(6)	1	0	0	1	1
34.03:de:6b:e4:49	6c:6c:d3:20:67:62	1	1	1	2375	8	1012	2861	0	0x0800	19018920401366_EVC6(6)	2375	0	0	2375	2375
74.4d:28:b8:61:e4	6c:6c:d3:20:67:62	1	1	1	8	14	1015	2861	0	0x0800	19018920401366_EVC6(6)	8	0	0	8	8
28.7b:09:af:54:ef	00:00:0c:9f:19:35	2	5	2	789385	0	1057	3125	0	0x0800	19018920401366_EVC4(4)	789385	0	0	789385	789385
74.4d:28:b8:61:e2	01:80:c2:00:00:00	5	4	5	9905	0	1480	1414	0	0x0027	19018920401366_EVC7(7)	9905	0	0	9905	9905
74.4d:28:b8:61:e2	01:80:c2:00:00:00	3	3	3	9905	0	2484	2554	0	0x0027	19018920401366_EVC3(3)	9905	0	0	9905	9905
74.4d:28:b8:61:e2	01:80:c2:00:00:00	3	3	3	9905	0	2520	2454	0	0x0027	19018920401366_EVC3(3)	9905	0	0	9905	9905
74.4d:28:b8:61:e2	01:80:c2:00:00:00	1	1	1	9905	0	2783	2705	0	0x0027	19018920401366_EVC2(2)	9905	0	0	9905	9905
74.4d:28:b8:61:e2	01:80:c2:00:00:00	1	1	1	9905	0	2914	2861	0	0x0026	19018920401366_EVC6(6)	9905	0	0	9905	9905
74.4d:28:b8:61:e2	01:80:c2:00:00:00	2	5	2	9905	1	3195	3125	0	0x0027	19018920401366_EVC4(4)	9905	0	0	9905	9905

Gambar 4. Duplicate MAC Address

Pada gambar 4 dapat terlihat *duplicate MAC address* terjadi pada setiap *VLAN ID* modem. *Source MAC* dan *destination MAC address* juga mengalami *duplicate MAC address* yang berarti menandakan salah satu *hardware user* mengalami *looping*.

4.3. Analisis Berdasarkan Trend Traffic PRTG (Paessler Router Traffic Grapher)

Semua sensor *traffic* yang ada pada *backhaul* dapat kita *monitoring* melalui *PRTG (Paessler Router Traffic Grapher)* yang memang secara spesifik digunakan untuk data penagihan *SLA* dan keperluan *troubleshooting network*. Dengan melihat *trend* grafik pada *monitoring PRTG* kita dapat menentukan *peak time traffic* dan *off-peak traffic*. Jika terjadi anomali penurunan *traffic* pada saat *peak time* tentunya akan berakibat penurunan *traffic* yang dapat kita lihat pada *monitoring PRTG*, seperti pada contoh gambar 5.

Gambar 5. Traffic Backhaul PRTG

Trend grafik backhaul PRTG yang putus pada gambar 5 dapat dengan jelas terlihat mengalami downtime, kejadian tersebut dapat mengindikasikan suatu masalah yang terjadi pada backhaul layer 2 switching.

Selain untuk data penagihan SLA (Service Level Agreement) monitoring PRTG juga dapat dijadikan acuan untuk proses troubleshooting seperti jitter ping, delay time, dan packet loss. Semua sensor yang ditarik ke dalam monitoring PRTG pada sisi user biasanya berupa jitter ping dan trend traffic VLAN, seperti pada gambar 6 dan gambar 7 berikut.

Gambar 6. Trend Packet Loss User

Trend grafik packet loss pada monitoring PRTG tentunya mengindikasikan performance network yang buruk. Tingginya persentase packet loss yang terjadi dapat dilakukan analisis baik itu pada saat peak time maupun off peak.

Gambar 7. Trend Traffic User

Pada gambar 7 trend grafik traffic user tentunya dapat dilakukan analisis yang menggambarkan penggunaan kapasitas bandwidth sudah digunakan secara maksimal.

4.4. Delay, Jitter, dan Packet Loss

Delay adalah waktu tunda sebuah proses pengiriman data dari satu titik sumber ke titik tujuan. Seperti di jalan raya jika dalam perjalanan ada hal-hal yang terjadi tidak terduga maka akan membuat proses perjalanan kita terganggu alias tidak sesuai target. Hal ini mungkin dalam perjalanan seringnya menjumpai lampu merah, maka perjalanan akan tertunda dari target yang diinginkan. Dalam jaringan internet sering terjadi delay pada saat media transmisi mengalami kestabilan[11].

Jitter adalah kumpulan dari semua *delay* yang terjadi selama proses data dikirimkan sampai dengan data diterima, *jitter* yang terjadi mendekati nol maka kecepatan jaringan sangat cepat[12]. Namun Sebaliknya jika tidak mendekati nol maka kecepatannya jelek dan akan terjadi kehilangan data dalam proses pengirimannya. Berikut ini contoh perhitungan nilai *jitter* pada persamaan (1) di bawah ini:

$$Jitter = \frac{\text{Total Variasi Delay}}{\text{Total Packet Diterima}}$$

Perhitungan Nilai *Jitter*.....(1)

Packet Loss adalah gagalnya proses transmisi data kepada alamat tujuan yang menyebabkan hilangnya beberapa data dalam proses pengiriman. Berikut merupakan perhitungan untuk *packet loss* pada persamaan (2) di bawah ini:

$$Packet Loss = \frac{(\text{Data Terkirim} - \text{Data Diterima})}{\text{Packet Data Terkirim}}$$

Perhitungan Nilai *Packet Loss*.....(2)

4.5. Penerapan COS Dengan Metode CBWFQ

CBWFQ (*Class Based Weighted Fair Queueing*) merupakan standar layanan penyediaan berdasarkan *class traffic* sesuai dengan kebutuhan *user*. Dalam membuat kelas tentunya dapat dilihat bagaimana *trend user experience* sehari-hari dalam menggunakan akses internet. Dari data tersebut kita dapat mengklasifikasikannya kedalam kelas-kelas yang nantinya akan kita buat dan mengatur *priority* di dalamnya. Dalam kelas yang sudah dibagi tersebut dapat ditentukan parameter seperti *bandwidth*, *weighting*, dan *packet size* yang dapat diatur sesuai kebutuhan *user*[13]. Untuk metode *CBWFQ* yang akan diterapkan nanti berada pada sisi *network L2 Hub PT. Iforte Global Internet*. Penerapan *class priority*, *weighting*, dan *packet size* diterapkan pada contoh gambar 8 dan gambar 9.

COS Parameters				COS Priority	
COS Type	Stateful Compression Mode	Number of Code Words	Max Buffer Size(Bytes)	Class of Services	Priority
CoS1	v44stateful	2048	10240	COS1	Interactive
CoS2	v44stateful	2048	10240	COS2	Interactive
CoS3	v44stateful	2048	10240	COS3	Streaming
CoS4	v44stateful	2048	10240	COS4	Bulk
CoS5	v44stateful	2048	10240	COS5	Bulk

Gambar 8. Penerapan COS Priority

Pada gambar 8 *COS Priority* dapat diterapkan melalui konfigurasi *NMS (Network Monitoring System) Jupiter*. Setiap urutan *COS Type* menandakan *priority VLAN* dan *packet* yang dapat diatur *Buffer Size* setiap *traffic packet* yang lewat.

Download Throughput SPI

Download Throughput L2_4M_TCBE

Any change in this SPI will affect

Virtual Network(s) 6

Terminal(s) 0

Service Plan(s) 0

IPGW(s) 13

General Expert Advanced

SPI Name: L2_4M_TCBE

Overall Scheduling Weight: 10

Differentiated QoS: ☒ Download Throughput Type: Traffic Class-based Best Effort

On Demand Real Time Enabled: ☐

Multi Stage FAP: ☐

Download Throughput Weight

CoS1 (%)	CoS2 (%)	CoS3 (%)	CoS4 (%)	CoS5 (%)
Interactive	Interactive	Streaming	Bulk	Bulk
50	50	100	50	50

Gambar 9. Penerapan Weighting COS

Penerapan *weighting COS* pada gambar 9 berfungsi untuk mengatur *label packet* yang akan diprioritaskan. Konfigurasi *weighting COS* dapat disesuaikan dengan keinginan *user packet layer 2* yang akan dijadikan prioritas di dalamnya.

Setelah dilakukan implementasi *CBWFQ* pada sisi *backhaul* terdapat penurunan *trend* grafik *packet loss*. Contoh gambar 10 data *trend* grafik *PRTG* berikut diambil selama kurun waktu 1 tahun terakhir.

Gambar 10. Trend Packet Loss Penerapan CBWFQ

Trend grafik *packet loss PRTG* selama 1 tahun pada gambar 10 menggambarkan penerapan *CBWFQ* (*Class Based Weighting Fair Queueing*) *trend packet loss* mengalami penurunan dimana performansi *network* semakin baik dan teratur sesuai dengan kebutuhannya.

4.6. Penerapan COS Dengan Metode PCQ

Algoritma *Per Connection Queue (PCQ)* merupakan algoritma yang digunakan untuk jumlah *user* yang banyak dan bisa gunakan untuk manajemen *bandwidth* dengan jumlah *user* yang tidak dapat diperkirakan, algoritma ini menyesuaikan banyaknya jumlah *user*. Penerapan manajemen *bandwidth* dengan algoritma *PCQ* (*Per Connection Queue*) pada sebuah jaringan merupakan cara yang terbaik untuk menangani jaringan yang jumlah *user* berubah secara dinamis atau tidak tetap. Algoritma *PCQ* (*Per Connection Queue*) merupakan penyempurnaan dari metode *SFQ* (*Stochastic Fairness Queuing*) yaitu dengan menyeimbangkan *traffic* dengan membuat beberapa *sub stream* (*sub queue*). Namun metode *Per Connection Queue (PCQ)* mempunyai penyempurnaan dibandingkan *SFQ*, jika *SFQ* menggunakan *Hash Algorithm* sebagai *Classifier*[14].

PCQ (Per Connection Queue) merupakan salah satu cara melakukan manajemen *bandwidth* yang cukup mudah, dimana *PCQ* bekerja dengan sebuah algoritma yang akan membagi *bandwidth* secara merata ke sejumlah *user* yang aktif. *PCQ* ideal diterapkan apabila dalam pengaturan *bandwidth* kita kesulitan dalam penentuan *bandwidth user* dalam jumlah banyak[15]. Berikut contoh gambar 11 & 12 implementasi metode *PCQ*.

Gambar 11. Penerapan Metode *PCQ* Pada Mikrotik User

Konfigurasi *queue PCQ* pada mikrotik router user dengan membedakan konfigurasi *PCQ download* dan *PCQ upload*. *IP source* dan *IP destination* dapat diatur sesuai dengan alokasi *bandwidth* yang user tentukan.

Gambar 12. Penerapan *PCQ Limiter*

Dalam konfigurasi *PCQ Limiter* dapat diatur *Queue Type* target *download* dan target *upload* serta mengatur *priority* di dalamnya. Segmentasi target *IP* dapat diatur sesuai dengan kebutuhan *user*.

Setelah dilakukan implementasi algoritma *PCQ*, pada contoh gambar 13 masih terdapat *trend packet loss* pada monitoring *PRTG* namun tidak lebih dari 10%. Jika dibandingkan dengan

metode algoritma *CBWFQ* yang memang sangat berdampak menurunkan *trend packet loss* karena di dalam metode *CBWFQ* banyak sekali hal yang dapat berpengaruh seperti *COS priority*, *weighting*, dan *packet size* yang dapat kita atur formulanya sesuai dengan kebutuhan *user*.

Gambar 13. *Trend Packet Loss Metode PCQ*

Trend packet loss metode *PCQ* secara bertahap mengalami penurunan namun tidak 100% *packet loss* dapat teratasi dengan baik, ini dikarenakan banyak sekali *packet* yang belum dilakukan *labeling* sesuai dengan penggunaan internet *user*.

Hasil perbandingan *jitter ping* dan *packet loss* dari algoritma *CBWFQ* dengan *PCQ* dijelaskan pada gambar 14 & 15. Perbandingan *jitter ping* *CBWFQ* lebih cocok diterapkan pada sisi *backhaul* karena terdapat penurunan *latency ping* yang lebih baik dibandingkan dengan algoritma *PCQ*, sedangkan untuk permasalahan perbandingan *packet loss* terdapat persentase pengurangan *packet loss* pada *VLAN monitoring*.

SAMPLE SITE	Metode CBWFQ		Metode PCQ	
	20-Aug-20	21-Aug-20	30-Aug-20	31-Aug-20
	20:00 - 23:59	00:00 - 04:00	20:00 - 23:59	00:00 - 04:00
NANGA LUAN	746 ms	534 ms	890 ms	768 ms
PARIGI	702 ms	612 ms	862 ms	792 ms
BEMUS	777 ms	533 ms	890 ms	765 ms
KADAM/OYIM	740 ms	576 ms	1201 ms	772 ms
L2-COLO-B3	798 ms	524 ms	796 ms	692 ms

Gambar 14. *Perbandingan Jitter Ping*

Perbandingan *jitter ping* pada gambar 14 menggambarkan metode *CBWFQ* secara signifikan menurunkan *latency ping* yang sebelumnya rata-rata 700 ms menjadi 500 ms. Metode *PCQ* meskipun dapat menurunkan *latency ping* namun tidak 100% untuk *trend packet loss* mengalami penurunan.

SAMPLE SITE	SER (%)	Downlink		Uplink		VLAN 2861 COS1		VLAN 2861 COS4	
		SQF	ModCod	SQF	ModCod	20-Aug-20	21-Aug-20	30-Aug-20	31-Aug-20
						20:00 - 23:59	00:00 - 04:00	20:00 - 23:59	00:00 - 04:00
NANGA LUAN	0,88	165	32APSK 5/6(26)	170	16APSK 1024K 9/10(215)	35%	4%	20%	7%
PARIGI	0,02	148	32APSK 7/9(88)	175	16APSK 1024K 9/10(215)	3%	0%	15%	2%
BEMUS	1,21	143	32APSK 3/4(24)	166	16APSK 1024K 9/10(215)	10%	0%	20%	10%
KADAM/OYIM	0,17	164	32APSK 5/6(26)	170	16APSK 1024K 9/10(215)	12%	2%	17%	8%
L2-COLO-B3	0,10	137	32APSK 25/36	170	16APSK 1024K 9/10(215)	5%	0%	18%	6%

PEAK HOURS	OFF PEAK HOURS
15%	-3%
-12%	-2%
-10%	-10%
-5%	-6%
-13%	-6%

Gambar 15. Perbandingan Nilai Packet Loss

Membandingkan nilai persentase *packet loss* pada saat *peak time* dan *off peak* menandakan semakin besar *traffic* yang lewat pada *layer 2 switching*, maka semakin besar pula persentase *packet loss* yang terjadi.

5. KESIMPULAN

Implementasi *Class Of Service (COS)* dengan membandingkan dua buah metode algoritma *CBWFQ (Class Based Weighted Fair Queueing)* & *PCQ (Per Connection Queue)*. *COS* dalam penelitian ini telah mampu menjalankan tugas utamanya yaitu dengan memberikan prioritas pada *VLAN monitoring SLA* dan layanan 2G yang membutuhkan *bandwidth realtime*. Hasil pengujiannya dua metode tersebut *CBWFQ* mampu menurunkan *packet loss* dan *high latency* yang terjadi apabila dibandingkan dengan metode *PCQ*, sehingga hilangnya *packet loss* berdampak pada performansi *network* yang maksimal.

Daftar Pustaka

- [1] I. O. Suzanti, A. Khozaimi, F. Teknik, and U. T. Madura, "Analisis Kinerja Efektifitas Pembagian Bandwidth Dan Quality Of Service (QoS) Pada Jaringan Kampus," vol. 3, no. 2, pp. 127–134, 2017.
- [2] A. V. Jha, A. N. Ghazali, B. Appasani, C. Ravariu, and A. Srinivasulu, "Reliability analysis of smart grid networks incorporating hardware failures and packet loss," *Rev. Roum. des Sci. Tech. Ser. Electrotech. Energ.*, vol. 65, no. 3–4, pp. 245–252, 2020.
- [3] Z. Ali, F. Naz, Javed, M. Qurban, M. Yasir, and S. Jehangir, "Analysis of VoIP over wired & wireless network with implementation of QoS CBWFQ & 802.11e," *Int. J. Comput. Netw. Inf. Secur.*, vol. 12, no. 1, pp. 43–39, 2020, doi: 10.5815/ijcnis.2020.01.05.
- [4] Y. A. Al-Sbou, "Wireless Networks Performance Monitoring Based On Passive-Active Quality Of Service Measurements," *Int. J. Comput. Networks Commun.*, vol. 12, no. 6, pp. 14–32, 2020, doi: 10.5121/ijcnc.2020.12602.
- [5] C. P. S. Cañar, J. J. T. Yépez, and H. M. R. López, "Performance of Reactive Routing Protocols DSR and AODV in Vehicular Ad-Hoc Networks Based on Quality of Service (Qos) Metrics," *Int. J. Eng. Adv. Technol.*, vol. 9, no. 4, pp. 2033–2039, 2020, doi: 10.35940/ijeat.c6608.049420.
- [6] N. S. R. Sherine Jenny, "Impact of Queuing Disciplines on the Performance of Multi-Class Traffic in a Network," *Inf. Technol. Ind.*, vol. 9, no. 1, pp. 698–703, 2021, doi: 10.17762/itii.v9i1.189.
- [7] M. Hanif and M. Kamisutara, "Sistem Monitoring Trafik Pada Mikrotik Berbasis App Mobile Dengan Notifikasi Telegram," *Netw. Eng. Res. Oper.*, vol. 6, no. 1, p. 1, 2021, doi: 10.21107/nero.v6i1.169.
- [8] A. Paramonov, N. Chistova, M. Makolkina, and A. Koucheryavy, *The method of forming the digital clusters for fifth and beyond telecommunication networks structure based on the quality of service*, vol. 12525 LNCS. Springer International Publishing, 2020.
- [9] F. X. Manggau, A. Latif, and Suwarjono, "E-Monitoring Microtic Network uses the Dude

- in Musamus University,” *J. Phys. Conf. Ser.*, vol. 1569, no. 2, 2020, doi: 10.1088/1742-6596/1569/2/022024.
- [10] A. Arnita and M. Farid, “Implementasi jaringan virtual private network dengan teknologi Multi Protocol Label Switching (MPLS),” vol. 5, no. 2, pp. 28–39, 2020.
- [11] A. G. Sleky and S. A. Safavi-naeini, “Handbook of Small Satellites,” *Handb. Small Satell.*, pp. 513–526, 2020, doi: 10.1007/978-3-030-36308-6.
- [12] S. Palinggi and Irmayani, “VSAT Bandwidth Efficiency on Satpath System,” *Int. J. Innov. Sci. Res. Technol.*, vol. 4, no. 12, pp. 1165–1174, 2019, [Online]. Available: <https://ijisrt.com/vsat-bandwidth-efficiency-on-satpath-system>.
- [13] M. Ulfah and A. S. Irtawaty, “P-51 Pengukuran Dan Analisa Quality Of Service (QOS) Jaringan Internet Di Gedung Terpadu Politeknik Negeri Balikpapan Measurement And Analysis Of The Internet Network Quality Of Perkembangan teknologi wireless yang digunakan oleh Kelancaran akses Intern,” pp. 351–357, 2020.
- [14] J. Barat and K. Kunci, “Implementasi Metode Per Connection Queue Dengan Access User Direct Mac Filtering Pada Jaringan Wireless,” pp. 240–249, 2020.
- [15] H. L. D. A. N. Non-leach, “Optimasi Network Lifetime Pada Jaringan Sensor Nirkabel Dengan Efisiensi Energi Menggunakan Teknik,” vol. XV, no. 1, 2020.