ANALYSIS SERVICE MPLS LDP WITH ROUTING DYNAMIC OSPF, QOS SERVICE ICMP.

Naufal Alauddin Yusuf

S1 Teknik Telekomunikasi 05 A – 17101030 – Divisi SDN

PROGRAM STUDI S1/D3 TEKNIK TELEKOMUNIKASI

FAKULTAS TEKNIK TELEKOMUNIKASI DAN ELEKTRO

INSTITUT TEKNOLOGI TELKOM PURWOKERTO

2019

BAB I
DASAR TEORI

A. MPLS (*Multi Protocol Label Switching*)

MPLS (*Multiprotocol Label Switching*) merupakan platform yang banyak dipilih untuk mentransport paket. MPLS memberikan solusi peningkatan performansi pada jaringan, dimana MPLS membuat jaringan lebih sederhana dengan cara menambahkan *header*/label pada paket sebagai identifikasi yang akan digunakan pada proses switching. Namun teknologi MPLS ini tetap tidak dapat memperhatikan kondisi jaringan. Sehingga apabila terjadi kongesti di jaringan tersebut maka tidak ada mekanisme tertentu untuk mengalihkannya ke jalur lain [1].

B. OSPF (*Open Short Path First*)

OSPF (*Open Shortest Path First*) adalah protokol yang digunakan dalam jaringan router sistem otonomi yang lebih besar dalam preferensi untuk *Routing Information Protocol* (RIP), protokol routing yang lebih tua yang dipasang di banyak jaringan perusahaan saat ini. Seperti RIP, OSPF ditunjuk oleh *Internet Engineering Task Force* (IETF) sebagai salah satu dari beberapa Protokol Interior Gateway [2]

C. *Wireshark*

Wireshark adalah program *Network Protocol Analyzer* alias penganalisa protokol jaringan yang lengkap. Program ini dapat merakam semua paket yang lewat serta menyeleksi dan menampilkan data tersebut se*detail* mungkin, misalnya postingan komentar kamu di blog atau bahkan *Username* dan *Password*.

Wireshark utamanya dibuat untuk *Administrator* Jaringan untuk dapat melacak apa yang terjadi didalam jaringan miliknya atau untuk memastikan jaringannya bekerja dengan baik, serta tidak ada yang melakukan hal hal buruk pada jaringan itu [3].

BAB II

MANFAAT DAN TUJUAN

A. MANFAAT

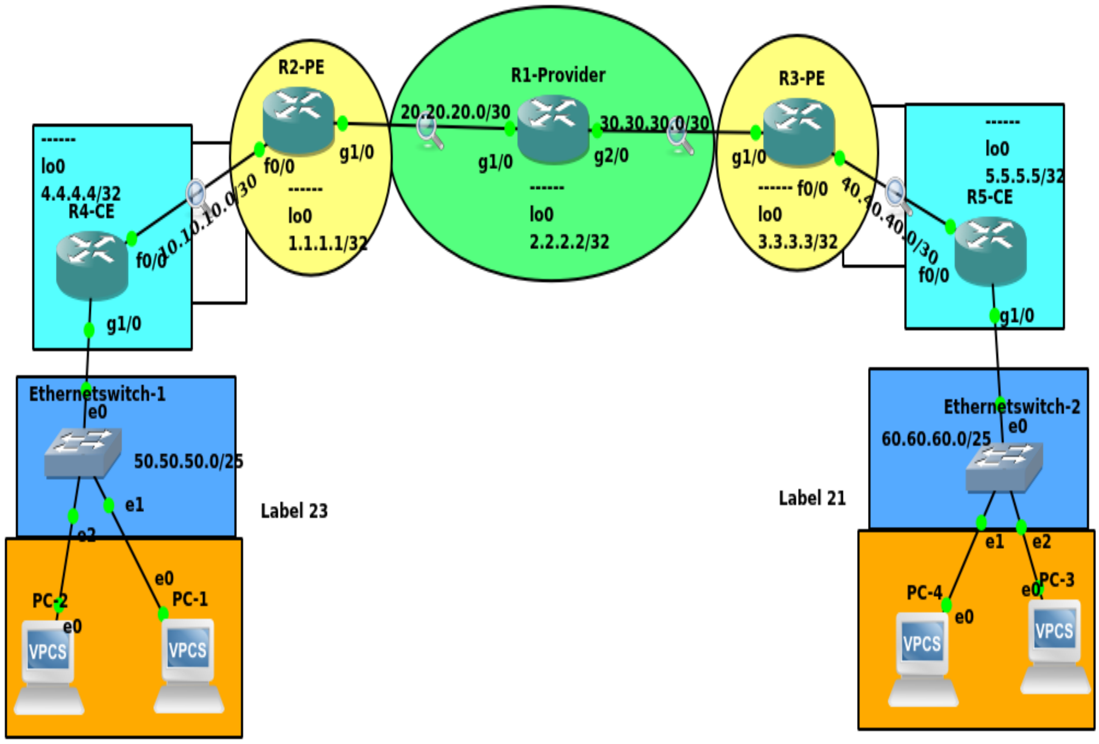
1) Meningkatkan kecepatan jaringan untuk konektivitas pertukaran data
2) Mengurangi banyak proses pengolahan data yang memerlukan resource disetiap hadwarenya, sehingga dapat mengoptimasi layanan jaringan yang dihasilkan.
3) Konsep keamanan yang sangat baik dibandingkan konsep keamanan jaringan konvensional.

B. TUJUAN

1) Mengimplementasikan teknologi MPLS LDP pada hadware cisco
2) Mengimplementasikan konsep routing dynamic OSPF
3) Menganalisa *service* ICMP dengan menggunakan *wireshark*

BAB III

KONFIGURASI PERANGKAT



Gambar 1.3.1 Topologi

Tabel 1.3.1 Konfigurasi pengalamatan IP

R1-*Provider*	R1-*Provider*> en R1-*Provider*#configure terminal **R1-*Provider*(config)#int lo0** R1-*Provider*(config-if)#ip add 2.2.2.2 255.255.255.255 R1-*Provider*(config-if)#no sh **R1-*Provider*(config-if)#int gig2/0** R1-*Provider*(config-if)#ip add 30.30.30.1 255.255.255.252 R1-*Provider*(config-if)#no sh **R1-*Provider*(config-if)#int gig0/0** R1-*Provider*(config-if)#ip add 20.20.20.1 255.255.255.252 R1-*Provider*(config-if)#no sh

R2-PE	R2-PE > en R2-PE#configure terminal **R2-PE(config)#int lo0** R2-PE(config-if)#ip add 1.1.1.1 255.255.255.255 R2-PE(config-if)#no sh **R2-PE(config-if)#int fa0/0** R2-PE(config-if)#ip add 10.10.10.1 255.255.255.252 R2-PE(config-if)#no sh **R2-PE(config-if)#int gig1/0** R2-PE(config-if)#ip add 20.20.20.2 255.255.255.252 R2-PE(config-if)#no sh
R3-PE	R3-PE > en R3-PE#configure terminal **R3-PE(config)#int lo0** R3-PE(config-if)#ip add 3.3.3.3 255.255.255.255 R3-PE(config-if)#no sh **R3-PE(config-if)#int fa0/0** R3-PE(config-if)#ip add 40.40.40.1 255.255.255.252 R3-PE(config-if)#no sh **R3-PE(config-if)#int gig1/0** R3-PE(config-if)#ip add 30.30.30.2 255.255.255.252 R3-PE(config-if)#no sh
R4-CE	R4-CE > en R4-CE#configure terminal **R4-CE(config)#int lo0** R4-CE(config-if)#ip add 4.4.4.4 255.255.255.255 R4-CE(config-if)#no sh **R4-CE(config-if)#int fa0/0** R4-CE(config-if)#ip add 10.10.10.2 255.255.255.252 R4-CE(config-if)#no sh **R4-CE(config-if)#int gig1/0** R4-CE(config-if)#ip add 50.50.50.1 255.255.255.128

	R4-CE(config-if)#no sh
R5-CE	R5-CE > en R4-CE#configure terminal **R4-CE(config)#int lo0** R4-CE(config-if)#ip add 5.5.5.5 255.255.255.255 R4-CE(config-if)#no sh **R4-CE(config-if)#int fa0/0** R4-CE(config-if)#ip add 40.40.40.2 255.255.255.252 R4-CE(config-if)#no sh **R4-CE(config-if)#int gig1/0** R4-CE(config-if)#ip add 60.60.60.1 255.255.255.128 R4-CE(config-if)#no sh

Tabel 1.3.2 Konfigurasi routing ospf

R1-*Provider*	R1-*Provider* >en R1-*Provider* #configure terminal **R1-*Provider*(config)#Router ospf 1** R1-*Provider*(config-router)#network 20.20.20.0 0.0.0.3 area 0 R1-*Provider*(config-router)#network 30.30.30.0 0.0.0.3 area 0 R1-*Provider*(config-router)#network 2.2.2.2 0.0.0.0 area 0 R1-*Provider*(config-router)#exit
R2-PE	R2-PE>en R2-PE#configure terminal **R2-PE(config)#Router ospf 1** R2-PE(config-router)#network 20.20.20.0 0.0.0.3 area 0 R2-PE(config-router)#network 10.10.10.0 0.0.0.3 area 0 R2-PE(config-router)#network 1.1.1.1 0.0.0.0 area 0 R2-PE(config-router)#exit
R3-PE	R3-PE>en R3-PE#configure terminal **R3-PE(config)#router ospf 1** R3-PE(config-router)#network 30.30.30.0 0.0.0.3 area 0

	R3-PE(config-router)#network 40.40.40.0 0.0.0.3 area 0 R3-PE(config-router)#network 3.3.3.3 0.0.0.0 area 0 R3-PE(config-router)#exit
R4-CE	R4-CE>en R4-CE#configure terminal **R4-CE(config)#router ospf 1** R4-CE(config-router)#network 10.10.10.0 0.0.0.3 area 0 R4-CE(config-router)#network 50.50.50.0 0.0.0.127 area 0 R4-CE(config-router)#network 4.4.4.4 0.0.0.0 area 0 R4-CE(config-router)#exit
R5-CE	R5-CE>en R5-CE#configure terminal **R5-CE(config)#router ospf 1** R5-CE(config-router)#network 40.40.40.0 0.0.0.3 area 0 R5-CE(config-router)#network 60.60.60.0 0.0.0.127 area 0 R5-CE(config-router)#network 5.5.5.5 0.0.0.0 area 0 R5-CE(config-router)#no auto R5-CE(config-router)#exit

Tabel 1.3.3 Konfigurasi DHCP Server

R4-CE	R4-CE>en R4-CE#configure terminal **R4-CE(config)#ip dhcp pool A** R4-CE(dhcp-config)#default-router 50.50.50.1 R4-CE(dhcp-config)#network 50.50.50.0 255.255.255.128
R5-CE	R5-CE>en R5-CE#configure terminal **R5-CE(config)#ip dhcp pool A** R5-CE(dhcp-config)#default-router 60.60.60.1 R5-CE(dhcp-config)#network 60.60.60.0 255.255.255.128

Tabel 1.3.4 Konfigurasi MPLS LDP

R1-*Provider*	R1-*Provider* >en **R1-*Provider* #configure terminal** R1-*Provider*(config)#mpls label *protocol* ldp R1-*Provider*(config)#mpls ldp session protection R1-*Provider*(config)#mpls ldp router-id loopback0 R1-*Provider*(config)#router ospf 1 R1-*Provider*(config-router)#mpls ldp autoconfiguration
R2-PE	R2-PE >en **R2-PE #configure terminal** R2-PE (config)#mpls label *protocol* ldp R2-PE (config)#mpls ldp session protection R2-PE (config)#mpls ldp router-id loopback0 R2-PE (config)#router ospf 1 R2-PE (config-router)#mpls ldp autoconfiguration
R3-PE	R3-PE >en **R3-PE #configure terminal** R3-PE (config)#mpls label *protocol* ldp R3-PE (config)#mpls ldp session protection R3-PE (config)#mpls ldp router-id loopback0 R3-PE (config)#router ospf 1 R3-PE (config-router)#mpls ldp autoconfiguration
R4-CE	R4-CE >en **R4-CE #configure terminal** R4-CE (config)#mpls label *protocol* ldp R4-CE (config)#mpls ldp session protection R4-CE (config)#mpls ldp router-id loopback0 R4-CE (config)#router ospf 1 R4-CE (config-router)#mpls ldp autoconfiguration
R5-CE	R5-CE >en **R5-CE #configure terminal** R5-CE (config)#mpls label *protocol* ldp

	R5-CE (config)#mpls ldp session protection R5-CE (config)#mpls ldp router-id loopback0 R5-CE (config)#router ospf 1 R5-CE (config-router)#mpls ldp autoconfiguration

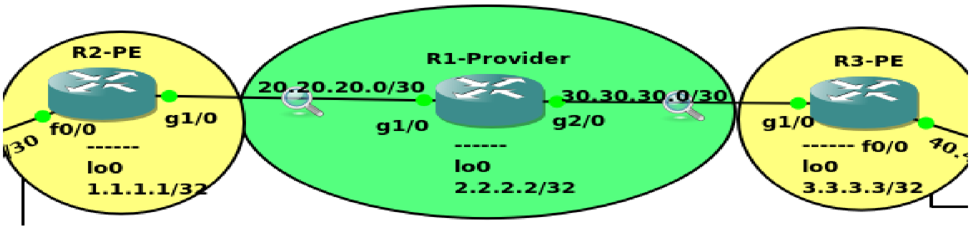
Tabel 1.3.5 Catatan konfigurasi dan keterangan

PERINTAH KONFIGURASI	KETERANGAN
Mpls label *protocol* ldp	(mengaktifkan ldp *protocol* pada router)
Mpls ldp session protection	"Ready –> Protecting". Ketika interface dihidupkan lagi session secara otomatis melakukan recovery.
Mpls ldp router-id loopback0	(menjadikan loopback router, menjadi router-id pada mpls ldp)
Mpls ldp autoconfiguration	untuk mengaktifkan fungsi konfigurasi mpls otomatis pada interface.
Int gig0/0	Memasuki interface gigabit ethernet0/0 dan melakukan berbagai macam konfigurasi yang bisa diterapkan.
Ip add 10.10.10.1 255.255.255.0	Menambahkan alamat IP pada salah satu interface
no sh / no shutdown	Melakukan perintah aktif terhadap interface
Router ospf 1	Konfigurasi routing dynamic OSPF dengan ID "1"
Network 10.10.10.0 0.0.0.255 area 0	Konfigurasi network 10.10.10.0 0.0.0.255 area 0, dimana **subnetmask** diubah menjadi **wildcard**.

RESULT CONFIGURATION

A. *NEIGHBOR DETAIL*

1) Router 1 *Provider*



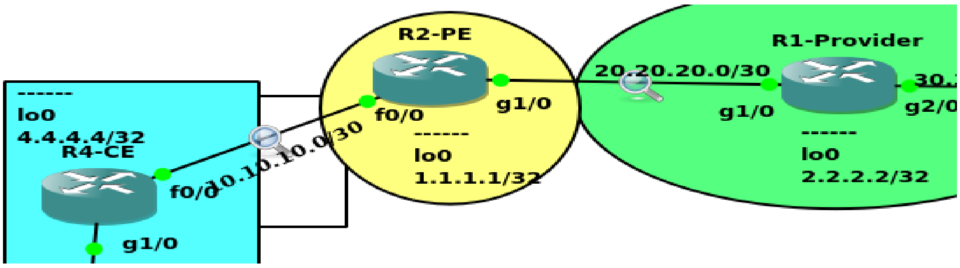
Tabel 1.3.2 R1 – *Provider*

```
R1-Provider#show mpls ldp neighbor detail
    Peer LDP Ident: 1.1.1.1:0; Local LDP Ident 2.2.2.2:0
        TCP connection: 1.1.1.1.646 - 2.2.2.2.34932
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 202/202; Downstream; Last TIB rev sent 22
        Up time: 02:44:32; UID: 1; Peer Id 0
        LDP discovery sources:
          Targeted Hello 2.2.2.2 -> 1.1.1.1, active, passive;
            holdtime: infinite, hello interval: 10000 ms
          GigabitEthernet1/0; Src IP addr: 20.20.20.2
            holdtime: 15000 ms, hello interval: 5000 ms
        Addresses bound to peer LDP Ident:
          1.1.1.1         10.10.10.1      20.20.20.2
```

```
    Peer LDP Ident: 3.3.3.3:0; Local LDP Ident 2.2.2.2:0
        TCP connection: 3.3.3.3.63268 - 2.2.2.2.646
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 200/202; Downstream; Last TIB rev sent 22
        Up time: 02:44:22; UID: 2; Peer Id 1
        LDP discovery sources:
          Targeted Hello 2.2.2.2 -> 3.3.3.3, active, passive;
            holdtime: infinite, hello interval: 10000 ms
          GigabitEthernet2/0; Src IP addr: 30.30.30.2
            holdtime: 15000 ms, hello interval: 5000 ms
        Addresses bound to peer LDP Ident:
          3.3.3.3         40.40.40.1      30.30.30.2
```

Tabel 1.3.3 R1 P – *Result Neighbor*

2) Router 2 *Provider Edge*



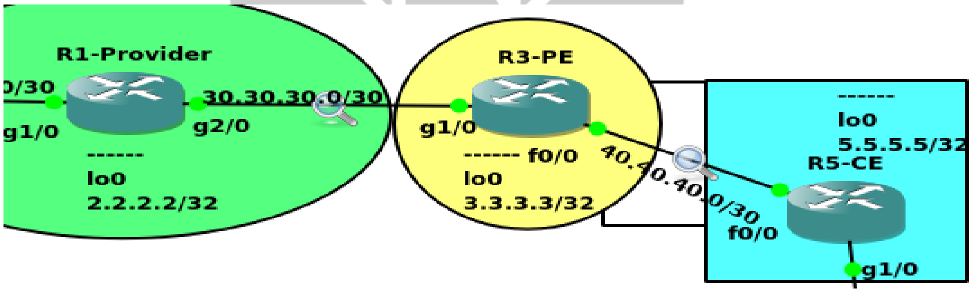
Tabel 1.3.4 R2 PE

```
R2-PE#show mpls ldp neighbor detail
    Peer LDP Ident: 2.2.2.2:0; Local LDP Ident 1.1.1.1:0
        TCP connection: 2.2.2.2.34932 - 1.1.1.1.646
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 239/238; Downstream; Last TIB rev sent 22
        Up time: 03:17:18; UID: 1; Peer Id 0
        LDP discovery sources:
          GigabitEthernet1/0; Src IP addr: 20.20.20.1
            holdtime: 15000 ms, hello interval: 5000 ms
          Targeted Hello 1.1.1.1 -> 2.2.2.2, active, passive;
            holdtime: infinite, hello interval: 10000 ms
        Addresses bound to peer LDP Ident:
          2.2.2.2         20.20.20.1      30.30.30.1
        Peer holdtime: 180000 ms; KA interval: 60000 ms; Peer state: estab
```

```
    Peer LDP Ident: 4.4.4.4:0; Local LDP Ident 1.1.1.1:0
        TCP connection: 4.4.4.4.56274 - 1.1.1.1.646
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 238/237; Downstream; Last TIB rev sent 22
        Up time: 03:17:07; UID: 2; Peer Id 1
        LDP discovery sources:
          Targeted Hello 1.1.1.1 -> 4.4.4.4, active, passive;
            holdtime: infinite, hello interval: 10000 ms
          FastEthernet0/0; Src IP addr: 10.10.10.2
            holdtime: 15000 ms, hello interval: 5000 ms
        Addresses bound to peer LDP Ident:
          4.4.4.4         10.10.10.2      50.50.50.1
```

Tabel 1.3.5 R2 PE – *Result Neighbor*

3) Router 3 *Provider Edge*



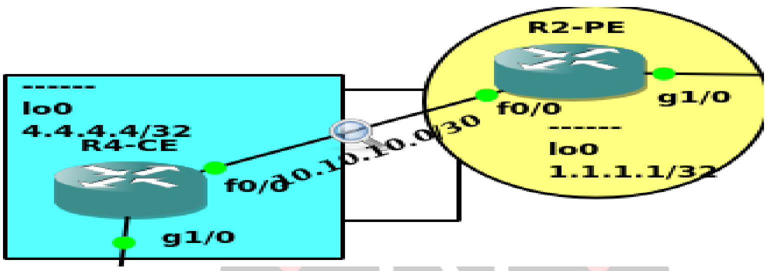
Tabel 1.3.6 R3 PE

```
R3-PE#show mpls ldp neighbor detail
    Peer LDP Ident: 2.2.2.2:0; Local LDP Ident 3.3.3.3:0
        TCP connection: 2.2.2.2.646 - 3.3.3.3.63268
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 205/202; Downstream; Last TIB rev sent 22
        Up time: 02:46:08; UID: 1; Peer Id 0
        LDP discovery sources:
          GigabitEthernet1/0; Src IP addr: 30.30.30.1
            holdtime: 15000 ms, hello interval: 5000 ms
          Targeted Hello 3.3.3.3 -> 2.2.2.2, active, passive;
            holdtime: infinite, hello interval: 10000 ms
        Addresses bound to peer LDP Ident:
          2.2.2.2         20.20.20.1      30.30.30.1
```

```
    Peer LDP Ident: 5.5.5.5:0; Local LDP Ident 3.3.3.3:0
        TCP connection: 5.5.5.5.57037 - 3.3.3.3.646
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 203/201; Downstream; Last TIB rev sent 22
        Up time: 02:46:01; UID: 2; Peer Id 1
        LDP discovery sources:
          FastEthernet0/0; Src IP addr: 40.40.40.2
            holdtime: 15000 ms, hello interval: 5000 ms
          Targeted Hello 3.3.3.3 -> 5.5.5.5, active, passive;
            holdtime: infinite, hello interval: 10000 ms
        Addresses bound to peer LDP Ident:
          5.5.5.5         40.40.40.2      60.60.60.1
```

Tabel 1.3.7 R3 PE – *Result Neighbor*

4) Router 4 *Customer Edge*

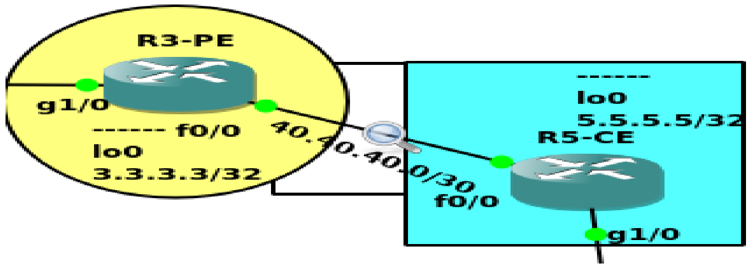


Tabel 1.3.8 CE

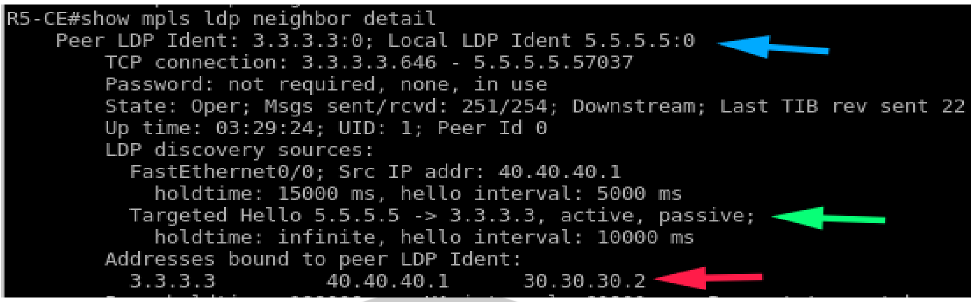
```
R4-CE#show mpls ldp neighbor detail
    Peer LDP Ident: 1.1.1.1:0; Local LDP Ident 4.4.4.4:0
        TCP connection: 1.1.1.1.646 - 4.4.4.4.56274
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 245/246; Downstream; Last TIB rev sent 22
        Up time: 03:23:18; UID: 1; Peer Id 0
        LDP discovery sources:
          FastEthernet0/0; Src IP addr: 10.10.10.1
            holdtime: 15000 ms, hello interval: 5000 ms
          Targeted Hello 4.4.4.4 -> 1.1.1.1, active, passive;
            holdtime: infinite, hello interval: 10000 ms
        Addresses bound to peer LDP Ident:
          1.1.1.1         10.10.10.1      20.20.20.2
```

Tabel 1.3.8 CE – *Result Neighbor*

5) **Router 5 *Customer Edge***



Tabel 1.3.9 CE



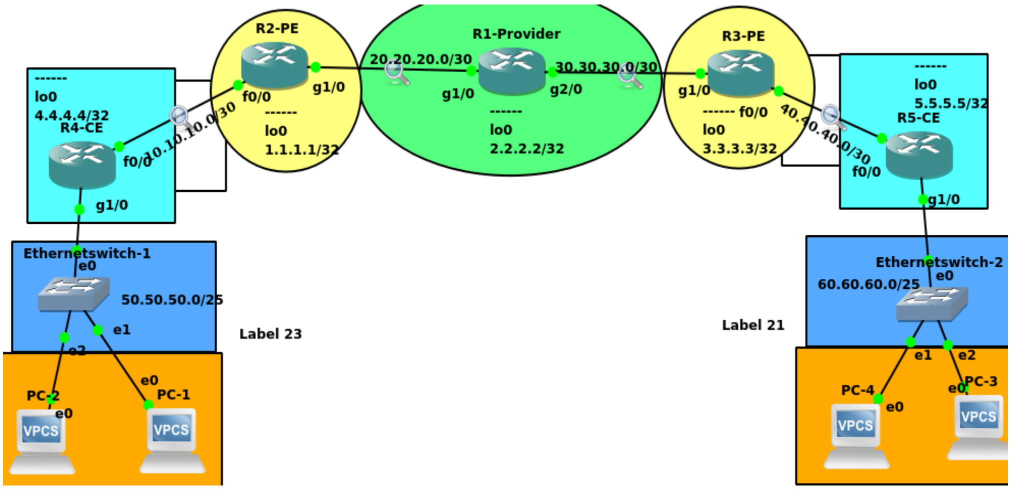
```
R5-CE#show mpls ldp neighbor detail
    Peer LDP Ident: 3.3.3.3:0; Local LDP Ident 5.5.5.5:0
        TCP connection: 3.3.3.3.646 - 5.5.5.5.57037
        Password: not required, none, in use
        State: Oper; Msgs sent/rcvd: 251/254; Downstream; Last TIB rev sent 22
        Up time: 03:29:24; UID: 1; Peer Id 0
        LDP discovery sources:
          FastEthernet0/0; Src IP addr: 40.40.40.1
            holdtime: 15000 ms, hello interval: 5000 ms
          Targeted Hello 5.5.5.5 -> 3.3.3.3, active, passive;
            holdtime: infinite, hello interval: 10000 ms
        Addresses bound to peer LDP Ident:
          3.3.3.3         40.40.40.1      30.30.30.2
```

Tabel 1.3.10 CE - *Neighbor*

Tabel 1.3.6 Keterangan

Kode Warna	**Keterangan**
Biru	Peer LDP Ident adalah LDP Lawan, Local IDP Ident adalah LDPnya sendiri
Hijau	Paket hello yang dikirimkan antara ldp untuk menandakan bahwa LDP lawannya active.
Merah	Merupakan jalur LDP yang menjadi jembatan untuk komunikasi antara MPLS di tiap router

B. *FORWARDING*



Tabel 1.3.11 Topologi

1) Router 1 *Edge*

R1-Provider#show mpls forwarding-table

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Label Switched	Outgoing interface	Next Hop
16	Pop Label	1.1.1.1/32	0	Gi1/0	20.20.20.2
17	Pop Label	10.10.10.0/30	0	Gi1/0	20.20.20.2
18	20	5.5.5.5/32	0	Gi2/0	30.30.30.2
19	19	4.4.4.4/32	0	Gi1/0	20.20.20.2
20	Pop Label	3.3.3.3/32	0	Gi2/0	30.30.30.2
21	22	60.60.60.0/25	612	Gi2/0	30.30.30.2
22	22	50.50.50.0/25	612	Gi1/0	20.20.20.2
23	Pop Label	40.40.40.0/30	0	Gi2/0	30.30.30.2

Tabel 1.3.12 R1 – *Result Forwarding*

2) Router 2 *Provider Edge*

R2-PE#show mpls forwarding-table

Local Label	Outgoing Label	Prefix or Tunnel Id	Bytes Label Switched	Outgoing interface	Next Hop
16	Pop Label	2.2.2.2/32	0	Gi1/0	20.20.20.1
17	Pop Label	30.30.30.0/30	0	Gi1/0	20.20.20.1
18	18	5.5.5.5/32	0	Gi1/0	20.20.20.1
19	Pop Label	4.4.4.4/32	0	Fa0/0	10.10.10.2
20	20	3.3.3.3/32	0	Gi1/0	20.20.20.1
21	21	60.60.60.0/25	5400	Gi1/0	20.20.20.1
22	Pop Label	50.50.50.0/25	6210	Fa0/0	10.10.10.2
23	23	40.40.40.0/30	0	Gi1/0	20.20.20.1

Tabel 1.3.13 R2 PE – *Result* Forwading

3) **Router 3 *Provider Edge***

```
R3-PE#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  2.2.2.2/32       0             Gi1/0      30.30.30.1
17         16         1.1.1.1/32       0             Gi1/0      30.30.30.1
18         17         10.10.10.0/30    0             Gi1/0      30.30.30.1
19         Pop Label  20.20.20.0/30    0             Gi1/0      30.30.30.1
20         Pop Label  5.5.5.5/32       0             Fa0/0      40.40.40.2
21         19         4.4.4.4/32       0             Gi1/0      30.30.30.1
22         Pop Label  60.60.60.0/25    6210          Fa0/0      40.40.40.2
23         22         50.50.50.0/25    5400          Gi1/0      30.30.30.1
```

Tabel 1.3.13 R3 PE – *Result Forwarding*

4) **Router 4 *Customer Edge***

```
R4-CE#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         20         3.3.3.3/32       0             Fa0/0      10.10.10.1
17         16         2.2.2.2/32       0             Fa0/0      10.10.10.1
18         Pop Label  1.1.1.1/32       0             Fa0/0      10.10.10.1
19         23         40.40.40.0/30    0             Fa0/0      10.10.10.1
20         17         30.30.30.0/30    0             Fa0/0      10.10.10.1
21         Pop Label  20.20.20.0/30    0             Fa0/0      10.10.10.1
22         18         5.5.5.5/32       0             Fa0/0      10.10.10.1
23         21         60.60.60.0/25    0             Fa0/0      10.10.10.1
```

Tabel 1.3.14 R4 PE – *Result* Forwading

5) **Router 5 *Customer Edge***

```
R5-CE#show mpls forwarding-table
Local      Outgoing   Prefix           Bytes Label   Outgoing   Next Hop
Label      Label      or Tunnel Id     Switched      interface
16         Pop Label  3.3.3.3/32       0             Fa0/0      40.40.40.1
17         16         2.2.2.2/32       0             Fa0/0      40.40.40.1
18         17         1.1.1.1/32       0             Fa0/0      40.40.40.1
19         18         10.10.10.0/30    0             Fa0/0      40.40.40.1
20         19         20.20.20.0/30    0             Fa0/0      40.40.40.1
21         Pop Label  30.30.30.0/30    0             Fa0/0      40.40.40.1
22         21         4.4.4.4/32       0             Fa0/0      40.40.40.1
23         23         50.50.50.0/25    0             Fa0/0      40.40.40.1
```

Tabel 1.3.15 R5 CE – *Result* Forwading

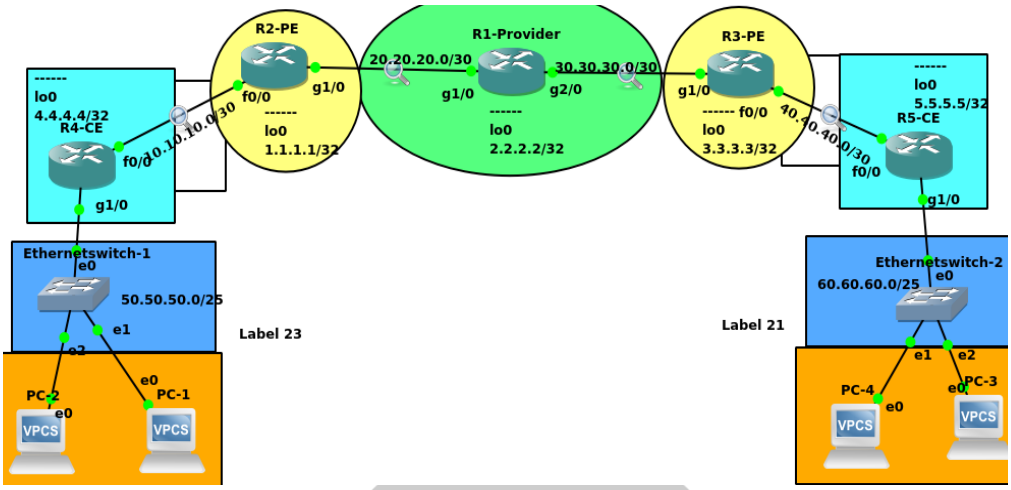
Tabel 1.3.7 Penjelasan

KODE WARNA	KETERANGAN
Hijau	IP Network client sisi A
Merah	IP Network sisi client B
Pop Label	Next-hop terakhir dari seluruh IP Tujuan

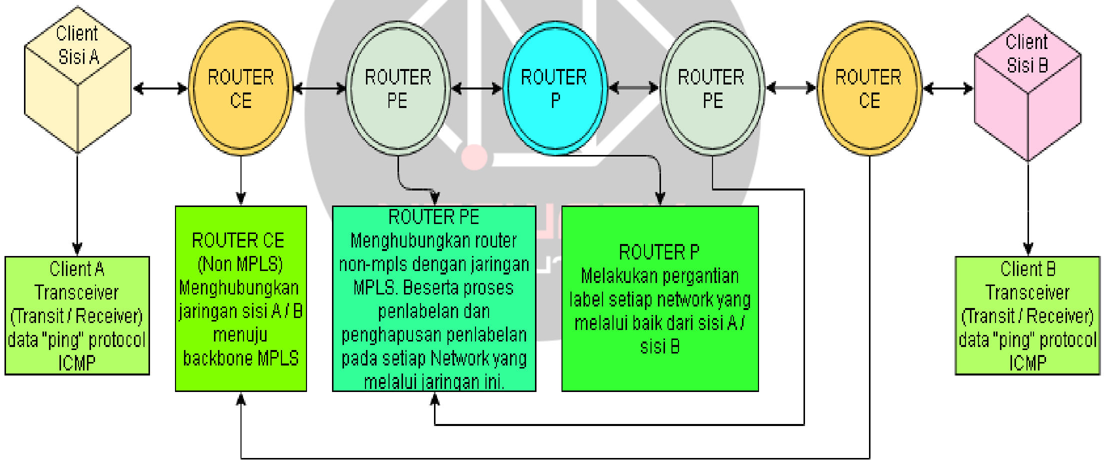
BAB IV

HASIL DATA DAN ANALISIS

A. Alur data dan penjelasan



Tabel 1.4.1 Topologi



Tabel 1.4.2 Alur Data Proses MPLS

- Client Sisi A (Label = 23) :
  1) PC 1 : 50.50.50.2 / 24
  2) PC 2 : 50.50.50.3 / 24
- Client sisi B (Label = 21) :
  1) PC 3 : 60.60.60.2 /24
  2) PC 4 : 60.60.60.3/24

- **Keterangan**

Proses penlabelan hanya terjadi saat melalui jalur komunikasi Router CE ↔ Router PE ↔ router P. Sedangkan saat Router CE ↔ Client Sisi A / B proses penlabelan tidak terjadi.

Router CE (A) akan mencantumkan network sisi client B kedalam *forwarding*-table yang dideklarasikan label 23 (local) dan label 21 (outgoing), sebaliknya **Router CE (B)** akan mencantumkan network sisi client A label 21 (local) dan label 23 (outgoing).

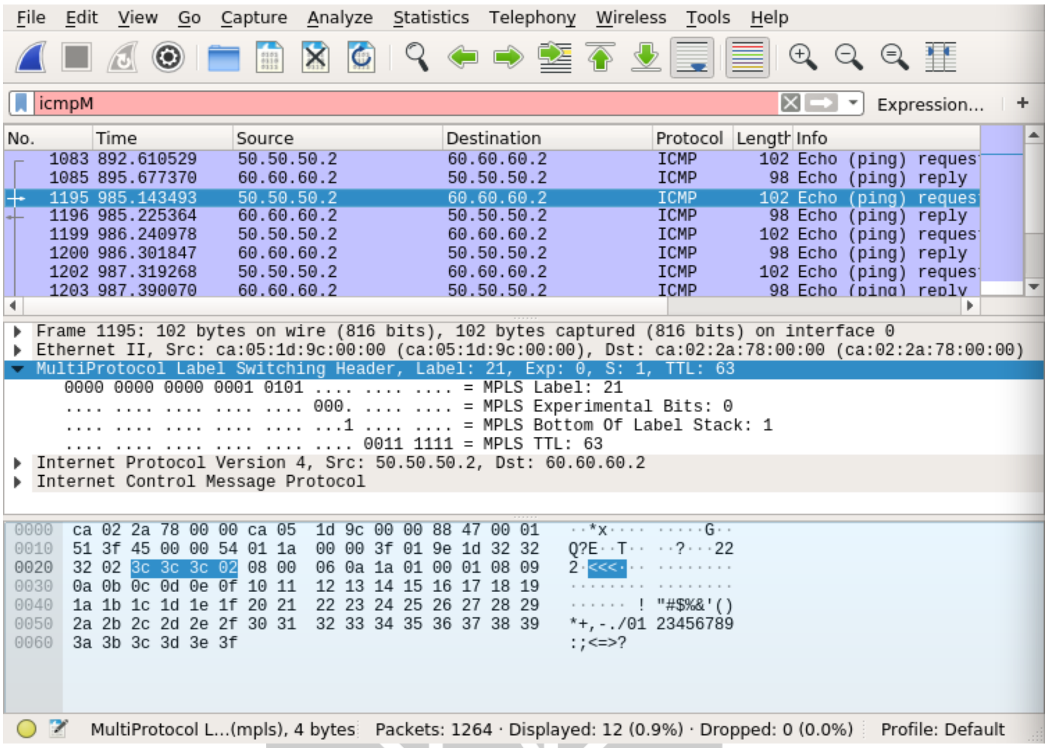
Router PE (A) akan mencantumkan network sisi client A, label 22 (local) dan Pop Label (outgoing), sedangkan untuk network sisi client B, label 21 (local) dan label 21 (outgoing). **Router PE (B)** akan mencantumkan network sisi, client A dengan label 23 (local) dan label 22 (outgoing), sedangkan untuk network sisi client B, label 22 (local) dan Pop Label (outgoing).

Router P (MPLS Backbone) Mencantumkan network sisi client A dan sisi client B, dimana untuk client sisi A label 21 (local) dan label 22 (outgoing) sedangkan untuk client sisi B label 22 (local) dan label 22 (outgoing).

Maka disimpulkan bahwa untuk network client sisi A memiliki label = 23 sedangkan network client sisi B memiliki label = 21. Sedangkan lainnya merupakan proses saat melakukan perubahan label hingga distribusi ke masing2 PE.

B. ANALISIS *SERVICE*

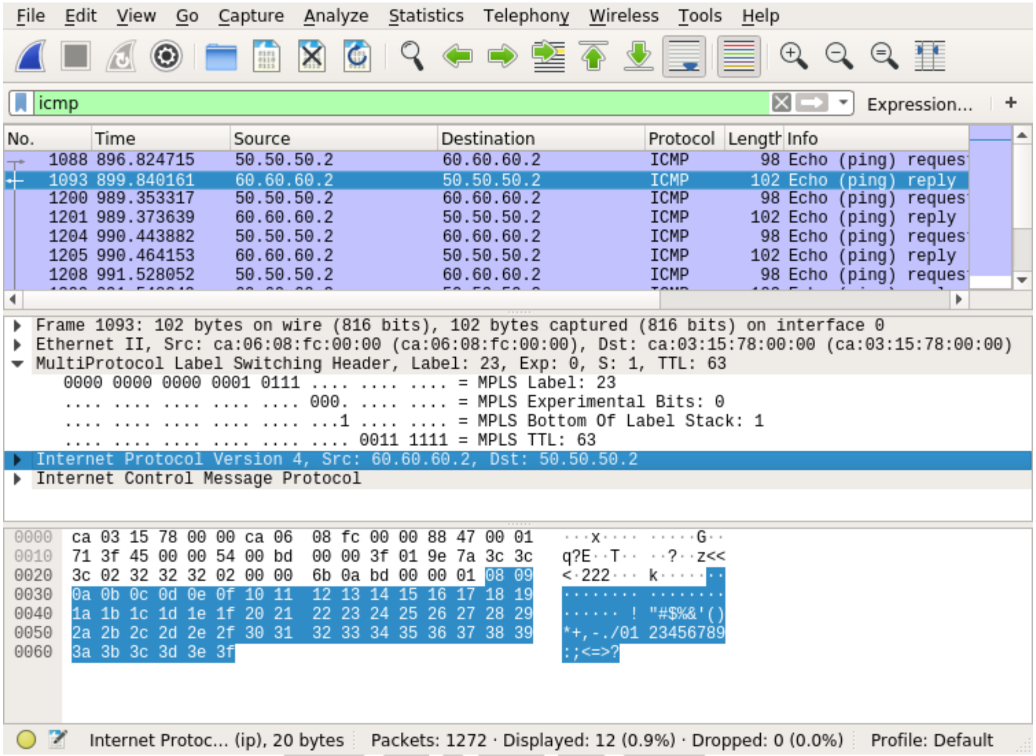
a) PC 1 → PC 3



Tabel 1.4.3 *Wireshark* PC1 → PC 3

PC 1 Melakukan ping terhadap→ PC 3 proses ini disebut dengan ping *request*, sebaliknya PC 3 menerima ping → PC 1 disebut dengan ping *reply*. Dalam proses ini ping menggunakan *protocol* "ICMP" Sehingga saat dilakukan anlisa mem-filter *service protocol* "ICMP" pada *wireshark*. Dalam hal ini didapatkan bahwa MPLS Label terjadi saat proses *request* ping yang dilakukan PC 1 terhadap PC3, ketika proses *reply* MPLS Label tidak terdekteksi karena tidak termasuk dalam *forwarding* table di Router PE (A).

b) PC 3 → PC 1



Tabel 1.4 PC3 → PC 1

PC 3 Melakukan ping balasan terhadap→ PC 1 proses ini disebut dengan ping *reply*. Dalam proses ini ping menggunakan *protocol* "ICMP" Sehingga saat dilakukan anlisa mem-filter *service protocol* "ICMP" pada *wireshark*. Dalam hal ini didapatkan bahwa MPLS Label terjadi saat proses *request* ping yang dilakukan PC 3 terhadap PC1, ketika proses *reply* MPLS Label tidak terdekteksi karena tidak termasuk dalam *forwarding* table di Router PE (B).

C. HASIL DATA

- *Throughput* : Bytes / *Time*span (*Delay* bytes / s)
- 1 bytes = 8 bit
- *Delay* = *Time*2 – *Time* 1

  Menggunakan isi nilai data “*Time*” untuk diisikan nilainya pada *time*2 dan *time*1
- Jitter = *Delay*2 – *Delay* 1

  Menggunakan isi nilai data “*Delay*” untuk diisikan nilainya pada *delay*2 dan *delay*1
- Total *delay* = Nilai keseluruhan *delay*
- Total jitter = Nilai keseluruhan Jitter
- Rata2 *delay* = Total *delay* / (Total *packet* – 1)
- Rata2 jitter = Total jitter / (Total jitter – 1)
- *Packet loss* $= \frac{y}{x}$ x 100

  Y = *Packet* data dikirim – *Packet* data diterima

  X = *Packet* data dikirim

Uji coba data dilakukan sebanyak 5 *request* dan 5 *reply* dengan beban pengiriman 102400 bytes

1) Sisi A → Sisi B

Result dari salah satu ping *request* sisi A → sisi B

Time

First packet:	2019-08-28 19:49:19
Last packet:	2019-08-28 19:52:32
Elapsed:	00:03:12

Capture

Hardware:	Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz (with SSE4.2)
OS:	Linux 4.18.0-25-generic
Application:	Dumpcap (Wireshark) 2.6.8 (Git v2.6.8 packaged as 2.6.8-1~ubuntu18.04.0)

Interfaces

Interface	Dropped packets	Capture filter	Link type	Packet size limit
-	Unknown	none	Ethernet	65535 bytes

Statistics

Measurement	Captured	Displayed	Marked
Packets	101	10 (9.9%)	—
Time span, s	192.578	6.260	—
Average pps	0.5	1.6	—
Average packet size, B	89	98	—
Bytes	9026	980 (10.9%)	0
Average bytes/s	46	156	—
Average bits/s	374	1.252	—

Tabel 1.4.5 *Capture* salah satu *packet* PC1 → PC3

Throughput: 9026 / 192.578 = 46,898 bytes/s = 46 bytes/s

46,898 x 8 = 374,958 bites/s

Packet loss : 0 %

No.	Time	Source	Destination	Protocol	Length	Info
67	139,946916	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xdb78, seq=1/256, ttl=64 (reply in 71)
69	141,947496	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xdd78, seq=2/512, ttl=64 (reply in 72)
71	143,001435	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xdb78, seq=1/256, ttl=59 (request in 67)
72	143,00149	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xdd78, seq=2/512, ttl=59 (request in 69)
73	143,948573	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xdf78, seq=3/768, ttl=64 (reply in 75)
75	144,020895	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xdf78, seq=3/768, ttl=59 (request in 73)
76	145,022665	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xe078, seq=4/1024, ttl=64 (reply in 77)
77	145,128585	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xe078, seq=4/1024, ttl=59 (request in 76)
78	146,129516	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xe278, seq=5/1280, ttl=64 (reply in 79)
79	146,207215	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xe278, seq=5/1280, ttl=59 (request in 78)

	time2	time1	delay	delay2	delay1	jitter
	141,947496	139,946916	2,000580	1,053939	2,000580	-0,946641
	143,001435	141,947496	1,053939	0,000055	1,053939	-1,053884
	143,00149	143,001435	0,000055	0,947083	0,000055	0,947028
	143,948573	143,00149	0,947083	0,072322	0,947083	-0,874761
	144,020895	143,948573	0,072322	1,001770	0,072322	0,929448
	145,022665	144,020895	1,001770	0,105920	1,001770	-0,89585
	145,128585	145,022665	0,105920	1,000931	0,105920	0,895011
	146,129516	145,128585	1,000931	0,077699	1,000931	-0,923232
	146,207215	146,129516	0,077699			
		total delay	6,260299		total jitter	-1,922881
		rata2 delay	0,062603		rata2 jitter	-0,01922881

Tabel 1.4.6 Perhitungan QoS Sisi A→ Sisi B

Dalam hal ini didapatkan bahwa rata-rata *delay* "0,0626003s = 60ms" dimana berdasarkan acuan dari *TIPHON* bahwa kategori *delay* <150ms = sangat bagus, sehingga *delay* dalam ujicoba ini dinyatakan sangat bagus.

Didapatkan bahwa rata-rata jitter "-0,01922881 s = -19.2881ms". Berdasarkan acuan dari *TIPHON* bahwa kategori jitter <0 = sangat bagus, sehingga jitter dalam ujicoba ini dinyatakan sangat bagus.

2) Sisi B → Sisi A

Result dari salah satu ping *reply* sisi B → sisi A

Time

First packet:	2019-08-28 19:49:25
Last packet:	2019-08-28 19:52:35
Elapsed:	00:03:10

Capture

Hardware:	Intel(R) Core(TM) i3-6006U CPU @ 2.00GHz (with SSE4.2)
OS:	Linux 4.18.0-25-generic
Application:	Dumpcap (Wireshark) 2.6.8 (Git v2.6.8 packaged as 2.6.8-1~ubuntu18.04.0)

Interfaces

Interface	Dropped packets	Capture filter	Link type	Packet size limit
-	Unknown	none	Ethernet	65535 bytes

Statistics

Measurement	Captured	Displayed	Marked
Packets	105	10 (9.5%)	—
Time span, s	190.048	6.187	—
Average pps	0.6	1.6	—
Average packet size, B	88	98	—
Bytes	9277	980 (10.6%)	0
Average bytes/s	48	158	—
Average bits/s	390	1.267	—

Tabel 1.4.7 *Capture* salah satu *packet* PC 3 → PC1

Troughput : 9277 / 190.578 = 48,678 bytes/s = 48 bytes/s

48,678 x 8 = 389,958 bites/s = 390 bites/s

Packet loss : 0 %

No.	Time	Source	Destination	Protocol	Length	Info
64	134,278315	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xdb78, seq=1/256, ttl=59 (reply in 74)
71	136,267091	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xdd78, seq=2/512, ttl=59 (reply in 75)
74	137,27967	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xdb78, seq=1/256, ttl=64 (request in 64)
75	137,279713	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xdd78, seq=2/512, ttl=64 (request in 71)
77	138,286526	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xdf78, seq=3/768, ttl=59 (reply in 78)
78	138,287252	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xdf78, seq=3/768, ttl=64 (request in 77)
79	139,375203	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xe078, seq=4/1024, ttl=59 (reply in 80)
80	139,375372	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xe078, seq=4/1024, ttl=64 (request in 79)
81	140,465253	50.50.50.2	60.60.60.2	ICMP	98	Echo (ping) request id=0xe278, seq=5/1280, ttl=59 (reply in 82)
82	140,465419	60.60.60.2	50.50.50.2	ICMP	98	Echo (ping) reply id=0xe278, seq=5/1280, ttl=64 (request in 81)

Time 2	Time 1	delay	Delay 2	Delay 1	Jitter
136,267091	134,278315	1,988776	1,012579	1,988776	-0,976197
137,27967	136,267091	1,012579	0,000043	1,012579	-1,012536
137,279713	137,27967	0,000043	1,006813	0,000043	1,00677
138,286526	137,279713	1,006813	0,000726	1,006813	-1,006087
138,287252	138,286526	0,000726	1,087951	0,000726	1,087225
139,375203	138,287252	1,087951	0,000169	1,087951	-1,087782
139,375372	139,375203	0,000169	1,089881	0,000169	1,089712
140,465253	139,375372	1,089881	0,000166	1,089881	-1,089715
140,465419	140,465253	0,000166			
	Total Delay	6,187104		Total jitter	-1,98861
	Rata2 delay	0,061871		Rata2 jitter	-0,0198861

Tabel 1.4.8 Perhitungan QoS PC 3 → PC1

Dalam hal ini didapatkan bahwa rata-rata *delay* “0,061871s = 60ms” dimana berdasarkan acuan dari *TIPHON* bahwa kategori *delay* <150ms = sangat bagus, sehingga *delay* dalam ujicoba ini dinyatakan sangat bagus.

Didapatkan bahwa rata-rata jitter “-0,0198861 s = -19.8861ms”. Berdasarkan acuan dari *TIPHON* bahwa kategori jitter <0 = sangat bagus, sehingga jitter dalam ujicoba ini dinyatakan sangat bagus.

BAB IV
KESIMPULAN

Dengan mengambil judul “Analysis *Service* MPLS LDP with routing Dynamic OSPF, QoS *service* ICMP” memiliki perbedaan yang cukup signifikan terutama saat membandingkan dengan saat masih menerapkan konsep jaringan konvensional. Dalam hal ini konsep MPLS sendiri sangat membantu ketika user melakukan transfer data dalam berbagai *service* seperti UDP,TCP,ICMP dan lain – lainnya dengan beban size yang terbilang cukup besar, hal yang dapat dibandingkan adalah dari kecepatan data yang dihasilkan.

DAFTAR PUSTAKA

[1] A. P. P. Wedda, D. I. R. M. dan R. M. , "IMPLEMENTASI DAN ANALISIS SOFT QoS (DIFFSERV) PADA JARINGAN MPLS," vol. 2, no. 2, p. 2916, 2015.

[2] A. "IMPLEMENTASI ROUTING *PROTOCOL* OPEN SHORTEST," vol. 8, no. 2, pp. 92-99, 2015.

[3] F. "Pengertian dan Fungsi *Wireshark*, sisi Hacker vs *Administrator* Jaringan," 23 Desember 2017. [Online]. Available: https://meretas.com/*wireshark*-adalah/. [Diakses 22 Agutus 2019].