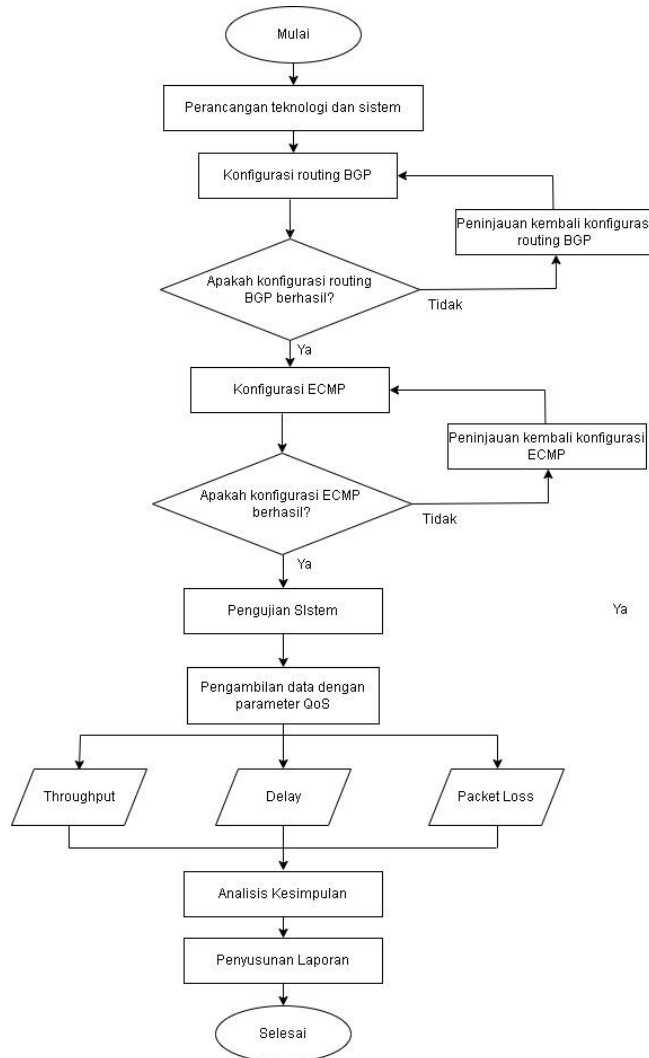


### BAB III METODE PENELITIAN

Penelitian dilakukan dengan menggunakan metode kuantitatif, dimana penyelesaian dari masalah penelitian dilakukan dengan mengumpulkan data yang kemudian diukur dan dianalisis sesuai dengan hasil pengujian dari penelitian ini.

#### 3.1 *FLOWCHART*

Berikut ini flowchart dari penelitian yang dilakukan digambarkan pada Gambar.3.2



**Gambar 3.2 *Flowchart* Penelitian**

Pada penelitian yang dilakukan ada beberapa proses yang harus dilakukan, mulai dari perancangan hingga pengambilan kesimpulan dan

penyusunan laporan. Tahapan pertama terdapat persiapan alat, mulai dari *hardware*, hingga *software* yang akan digunakan. Tahap selanjutnya perancangan sistem, perancangan sistem menentukan topologi yang akan digunakan, pengujian seperti apa yang akan dilakukan. Tahapan ketiga melakukan konfigurasi *routing BGP* pada topologi yang sudah dirancangan sebelumnya. Tahap selanjutnya mengimplementasikan *load balancing* yang digunakan, yaitu *ECMP (Equal Cost Multi Path)* juga dilakukan. Selanjutnya dilakukan pengujian sistem dilanjutkan dengan pengambilan data dengan tiga parameter pengujian yaitu *throughput*, *delay* dan *packet loss*. Pengujian juga dilakukan pada jaringan yang tidak diimplementasikan metode *load balancing* dari implementasi tersebut diambil 3 parameter pengujian yaitu *throughput*, *delay* dan *packet loss*. Data dari tiga parameter yang didapatkan akan dibandingkan, dianalisa dan diambil kesimpulan yang kemudian akan dijadikan bahan penyusunan laporan.

### 3.2 PERANGKAT SIMULASI

Penelitian menggunakan perangkat keras (*hardware*) dan perangkat lunak (*software*) yang digunakan untuk simulasi dan pengujian pada penelitian:

#### 3.2.1 Perangkat Keras (*Hardware*)

Pada penelitian digunakan *hardware* berupa 1 buah laptop untuk menjalankan *software* simulasi, berikut ini spesifikasi dari *hardware* yang digunakan, terlampir pada Tabel 3.1 berikut.

**Tabel 3. 1 Spesifikasi Laptop**

Sistem Operasi	Windows 11 Pro 64-bit
<i>Processor</i>	11 <sup>th</sup> Gen Intel® Core™ i7-1165G7 8 CPU @ 2.80GHz
<i>RAM</i>	16 GB
<i>Storage</i>	500 GB

#### 3.2.2 Perangkat Lunak (*Software*)

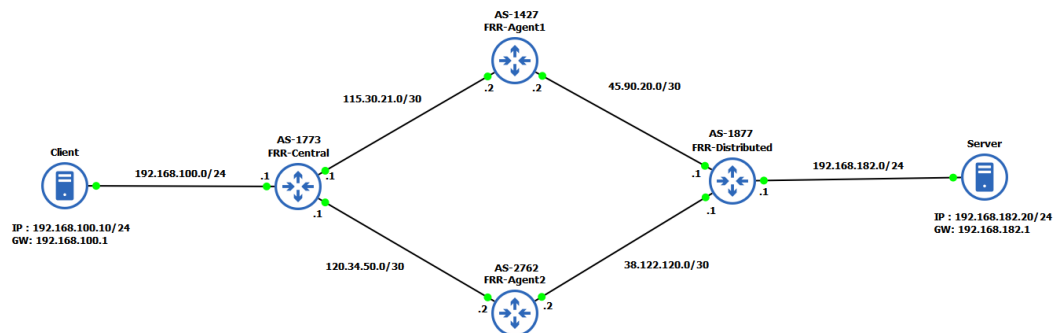
Selain *hardware* digunakan juga *software* sebagai penunjang dari penelitian ini. Berikut ini *software* yang digunakan:

- a. *GNS3*, digunakan untuk mensimulasikan topologi jaringan, aplikasi *GNS3* diinstall pada *hardware / laptop* yang digunakan untuk pengujian.

- b. *Wireshark*, berfungsi untuk menangkap *traffic* pada topologi jaringan yang diuji.
- c. *FRRouting*, berfungsi sebagai *router open network*, digunakan untuk implementasi *routing BGP*.

### 3.3 RANCANGAN PENELITIAN

Rancangan dari penelitian berisi topologi dan informasi *device* yang ada topologi tersebut. Pada penelitian yang dilakukan menggunakan topologi jaringan *full mesh* dengan 4 *FRRouter* yang berfungsi menghubungkan jaringan dengan *network* yang berbeda dengan 1 *FRRouter* berperan sebagai *router central* dengan AS 1773 *router central* akan berperan sebagai *FRR core* yang akan mendistribusikan trafik ke jalur *router agent 1* dan *router agent 2*, 1 *FRouter* sebagai *router distributed* dengan AS 1877 memiliki peran yang sama seperti *router central* akan berperan sebagai *FRR core* yang akan menerima distribusi trafik dari kedua jalur, dan 2 *FRRouter* lainnya sebagai *router agent* kedua router agent akan berperan sebagai jalur distribusi trafik. Topologi tersebut juga dilengkapi dengan *client* dan *server* yang berperan sebagai *endpoint* dan masing-masing menggunakan sistem operasi *Ubuntu*. Simulasi penelitian ini menggunakan *ethernet* karena keterbatasan dari *software* simulasi *GNS3* yang digunakan hanya *support* untuk *ethernet*.



**Gambar 3.3 Topologi Jaringan**

Selain topologi jaringan pada Gambar 3.3, berikut ini dilampirkan juga informasi *device* yang ada pada topologi tersebut pada Tabel 3.2, yang berisi informasi seperti *port*, *IP address*, *Autotomous System Number*, dan *Gateway*, dan kemana *device* tersebut terhubung.

**Tabel 3. 2 Konfigurasi Topologi**

<i>Device</i>	<i>Interface</i>	<i>IP Address</i>	<i>AS/GW</i>	<i>Connect to</i>
<i>Server</i>	eth0	192.168.100.10/24	192.168.100.1	<i>frr-central</i>
<i>Client</i>	eth0	192.168.182.20/24	192.168.182.1	<i>frr-distributed</i>
<i>FRR-Central</i>	lo	10.50.10.10/32	1773	
	eth0	115.30.21.1/30		<i>FRR-Agent1</i>
	eth2	120.34.50.1/30		<i>FRR-Agent3</i>
	eth5	192.168.100.1/24		<i>Server</i>
<i>FRR-Agent1</i>	lo	10.60.10.10/32	1427	
	eth0	115.30.21.2/30		<i>frr-central</i>
	eth1	45.90.20.2/30		<i>frr-distributed</i>
<i>FRR-Agent2</i>	lo	10.80.10.10/32	2762	
	eth2	120.34.50.2/30		<i>frr-central</i>
	eth3	38.22.120.2/30		<i>frr-distributed</i>
<i>FRR-Distributed</i>	lo	10.100.10.10/32	1877	
	eth1	45.90.20.1/30		<i>FRR-Agent1</i>
	eth3	38.22.120.1/30		<i>FRR-Agent3</i>
	eth5	192.168.182.1/24		<i>Server</i>

Pada Tabel 3.2 terdapat *client* dan *server* yang keduanya menggunakan sistem operasi *Ubuntu*. Perangkat *FRRouter* berperan sebagai *Router Central*, *Router Distributed*, *Router Agent 1*, *Router Agent 2*, dan *Router agent 3*.

### **3.4 KONFIGURASI PADA *FRR-CENTRAL***

Konfigurasi yang pertama dilakukan adalah konfigurasi pada *FRR-Central*, berikut konfigurasinya pada Gambar 3.4

```

hostname frr
service integrated-vtysh-config
!
interface eth0
 ip address 115.30.21.1/30
!
interface eth2
 ip address 120.34.50.1/30
!
interface eth5
 ip address 192.168.100.1/24
!
interface lo
 ip address 10.50.10.10/32
!
router bgp 1773
 bgp bestpath as-path multipath-relax
 neighbor central peer-group
 neighbor central remote-as external
 neighbor 115.30.21.2 peer-group central
 neighbor 120.34.50.2 peer-group central
!
 address-family ipv4 unicast
  network 10.50.10.10/32
  network 192.168.100.0/24
  maximum-paths 2
 exit-address-family
!
line vty
!
end
frr# █

```

**Gambar 3. 4 Konfigurasi pada *FRR-Central***

Pada konfigurasi tersebut pada *interface ethernet 0* diberikan *IP address* 115.30.21.1/30 *interface ethernet 0* nantinya akan mengarah ke *router agent 1*. Pada *interface ethernet 2* dikonfigurasi *IP* 120.34.50/30 *interface ethernet 2* akan terhubung dengan *router agent 2*. Pada *interface ethernet 5* dikonfigurasi *IP address* 192.168.100.1 yang akan terhubung dengan *client*. Pada *loopback interface* diberikan *IP address* 10.50.10.10/32 sebagai *IP loopback*. Dilakukan pengaturan *routing BGP* dengan memberikan *FRR-Central AS* 1773. Lalu perintah *bgp bestpath as-path mulipath-relax* yang memungkinkan *router* berbagi jalur trafik yang memiliki *AS* yang berbeda. Dibuat *neighbor group* dengan nama *central neighbor group* ini akan membuat konfigurasi *BGP* lebih sederhana. Perintah *neighbor central remote-as external* konfigurasi ini digunakan agar *router* menganggap *neighbor* sebagai *external*. Lalu diinputkan *IP neighbor* pada grup *neighbor central*. Perintah *address-family ipv4 unicast* digunakan untuk

mengkonfigurasi agar *router* bisa berkomunikasi dengan *host* lain pada jaringan *ipv4*. Dikonfigurasi juga *network* yang terhubung dengan *router central*. Perintah *maximum-paths 2* digunakan untuk mengatur maksimal jalur yang akan digunakan.

### 3.5 KONFIGURASI PADA *FRR-DISTRIBUTED*

Konfigurasi dilakukan juga pada *router distributed*, konfigurasi pada *router distributed* hampir sama dengan konfigurasi pada *router central* berikut pada Gambar 3.5

```
hostname frr
service integrated-vtysh-config
!
interface eth1
 ip address 45.90.20.1/30
!
interface eth3
 ip address 38.22.120.1/30
!
interface eth5
 ip address 192.168.182.1/24
!
interface lo
 ip address 10.100.10.10/32
!
router bgp 1877
 bgp bestpath as-path multipath-relax
 neighbor central peer-group
 neighbor central remote-as external
 neighbor 38.22.120.2 peer-group central
 neighbor 45.90.20.2 peer-group central
!
 address-family ipv4 unicast
  network 10.100.10.10/32
  network 192.168.182.0/24
  maximum-paths 2
 exit-address-family
!
line vty
!
end
frr#
```

**Gambar 3. 5 Konfigurasi pada *FRR-Distributed***

*IP address* dikonfigurasi pada *interface ethernet1*, *ethernet3*, dan *ethernet 5*. Dengan *interface ethernet 1* dan *ethernet 3* terhubung ke *router agent 1* dan *router agent 2*, *interface ethernet 5* terhubung ke *server*. Dikonfigurasi juga pada *interface loopback* dengan *IP 10.100.10.10/32*. Dikonfigurasi juga *routing BGP* dengan *AS 1877* pada *router distributed*. Perintah *bgp bestpath as-*

*path multipath-relax* yang memungkinkan *router* berbagi jalur trafik yang memiliki AS yang berbeda. Dikonfigurasi juga *group neighbor* yang sama dengan *router central*. Perintah *neighbor central remote-as external* konfigurasi ini digunakan agar *router* menganggap *neighbor* sebagai *external*. Diinputkan IP *neighbor router* yang terhubung dengan *router distributed*. Perintah *address-family ipv4 unicast* untuk mengkonfigurasi agar *router* bisa berkomunikasi dengan *host* lain pada jaringan *ipv4*. Dikonfigurasi juga *network* yang terhubung dengan *router central* dan maksimal jalur yang akan digunakan.

### 3.6 KONFIGURASI PADA FRR-AGENT1

*FRR-Agent1* berperan sebagai salah satu jalur topologi pada penelitian ini. *Router FRR-Agent1* memiliki AS 1427, dengan konfigurasi sebagai berikut pada Gambar 3.6.

```
frr-ubuntu@ubuntu:~$ cat /etc/frr/frr.conf
hostname frr
service integrated-vtysh-config
!
interface eth0
 ip address 115.30.21.2/30
!
interface eth1
 ip address 45.90.20.2/30
!
interface lo
 ip address 10.60.10.10/32
!
router bgp 1427
 bgp bestpath as-path multipath-relax
 neighbor central peer-group
 neighbor central remote-as external
 neighbor 45.90.20.1 peer-group central
 neighbor 115.30.21.1 peer-group central
!
 address-family ipv4 unicast
  network 10.60.10.10/32
 exit-address-family
!
line vty
!
end
frr#
```

**Gambar 3. 6 Konfigurasi FRR-Agent1**

Pada *interface ethernet 0* dikonfigurasi *IP address* 115.30.21.2/30 agar bisa terhubung dengan *router central*, untuk *interface ethernet 1* dikonfigurasi *IP* 45.90.20.2/30 untuk terhubung dengan *router distributed*, *interface loopback* diinputkan dengan *IP loopback* 10.60.10.10/32. *Router agent1* dikonfigurasi

dengan AS 1427. Dengan konfigurasi agar *router agent* bisa berbagi jalur trafik dengan AS yang berbeda, juga konfigurasi *neighbor group central* agar *router agent1* bisa berbagi informasi dengan tetangga *router central* dan *router distributed*, diinputkan juga *network localhost*.

### 3.7 KONFIGURASI PADA FRR-AGENT2

*FRR-Agent2* berperan sebagai salah satu jalur topologi pada penelitian ini, dengan konfigurasi berikut pada Gambar 3.7.

```
hostname frr
service integrated-vtysh-config
!
interface eth2
 ip address 120.34.50.2/30
!
interface eth3
 ip address 38.22.120.2/30
!
interface lo
 ip address 10.80.10.10/32
!
router bgp 2762
 bgp bestpath as-path multipath-relax
 neighbor central peer-group
 neighbor central remote-as external
 neighbor 38.22.120.1 peer-group central
 neighbor 120.34.50.1 peer-group central
!
 address-family ipv4 unicast
  network 10.80.10.10/32
 exit-address-family
!
line vty
!
end
frr#
```

**Gambar 3.7 Konfigurasi FRR-Agent2**

*Interface ethernet 2* diinputkan IP 120.34.50.2/30, *interface ethernet 3* diinputkan IP 38.22.120.2/30, *interface loopback* diinputkan IP loopback 10.80.10.10/32. *Router agent 2* memiliki AS 2762 dengan *neighbor 38.22.120.1*



yang terhubung ke *router distributed* dan 120.34.50.1 yang terhubung ke *router central*.

### 3.8 KONFIGURASI *LOAD BALANCING ECMP*

Berikut konfigurasi load balancing ECMP digambarkan pada Gambar 3.8 Dijelaskan juga fungsi dari masing masing command pada penjelasan berikut

```
router bgp 1773
  bgp bestpath as-path multipath-relax
  neighbor central peer-group
  neighbor central remote-as external
  neighbor 115.30.21.2 peer-group central
  neighbor 120.34.50.2 peer-group central
  !
  address-family ipv4 unicast
    network 10.50.10.10/32
    network 192.168.100.0/24
    maximum-paths 2
  exit-address-family
  !
```

**Gambar 3. 8 Konfigurasi *Load Balancing ECMP***

Konfigurasi *load balancing ECMP* menggunakan perintah *bgp bestpath as-path multipath-relax*, perintah tersebut membuat *router* berbagi beban pada beberapa jalur walaupun *AS* nya berbeda. Perintah *maximum-paths 2* juga digunakan sebagai konfigurasi *load balancing*, perintah tersebut digunakan untuk mengatur jalur maksimal yang digunakan.

Kekurangan dari *load balancing ECMP* pada *FRRouting* adalah belum tersedianya konfigurasi untuk mengatur besarnya beban trafik pada masing-masing jalur, sehingga trafik yang didistribusikan dikirim secara acak.

### 3.9 KONFIGURASI *CLIENT*

Setelah dilakukan konfigurasi pada *router*, dilakukan juga konfigurasi pada *client* karena *client* sendiri nantinya akan berperan sebagai pengirim data saat dilakukan pengujian pengiriman data dengan ukuran yang bervariasi dari mulai ukuran 100 MB, 2456 MB, 512 MB dan 1 GB dengan melakukan konfigurasi Gambar 3.9.

```
GNU nano 2.9.3 /etc/netpla
# This file describes the network interfaces
# For more information, see netplan(5).
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    eth0:
      dhcp4: false
      addresses: [192.168.100.10/24]
      gateway4: 192.168.100.1
    eth1:
      dhcp4: false
```

**Gambar 3. 9 Konfigurasi Client**

Karena yang terhubung dengan *router central eth0*, maka pada *eth0* diinputkan *IP address* 192.168.100.10/24 sebagai *IP* dari *client*, dan dikonfigurasi juga *IP gateway* dengan *IP* 192.168.100.1.

### 3.10 KONFIGURASI SERVER

Setelah dilakukan konfigurasi pada *router*, dilakukan juga konfigurasi pada *server* sama seperti *client*, hanya saja *server* akan berperan menerima *data* yang dikirimkan dari sisi *client* pada Gambar 3.10

```
# This file describes the network interfaces
# For more information, see netplan(5).
network:
  version: 2
  renderer: NetworkManager
  ethernets:
    eth0:
      dhcp4: false
      addresses : [192.168.182.20/24]
      gateway4 : 192.168.182.1
    eth1:
      dhcp4: false
```

**Gambar 3. 10 Gambar Konfigurasi Server**

konfigurasi dilakukan juga pada *IP address* dengan *IP* 192.168.182.20/24, dan *IP gateway* 192.168.182.20. Pada penelitian yang dilakukan menggunakan sistem operasi *Ubuntu 18.04*, konfigurasi dilakukan pada *netplan* pada *file yaml*.

### 3.11 VERIFIKASI ROUTING BGP

Setelah dilakukan konfigurasi pada *routing BGP*, dilakukan verifikasi pada masing-masing *router* pada Gambar 3.11

```
frr# show ip bgp
BGP table version is 9, local router ID is 10.50.10.10, vrf id 0
Default local pref 100, local AS 1773
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
              i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @N.NN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network        Next Hop        Metric LocPrf Weight Path
*> 10.50.10.10/32  0.0.0.0          0         32768 i
*> 10.60.10.10/32  115.30.21.2      0         0 1427 i
* 10.60.10.10/32  120.34.50.2      0         0 2762 1877 1427 i
* 10.80.10.10/32  115.30.21.2      0         0 1427 1877 2762 i
*> 10.80.10.10/32  120.34.50.2      0         0 2762 i
*= 10.100.10.10/32 115.30.21.2      0         0 1427 1877 i
*> 10.100.10.10/32 120.34.50.2      0         0 2762 1877 i
*> 192.168.100.0/24 0.0.0.0          0         32768 i
*= 192.168.182.0/24 115.30.21.2      0         0 1427 1877 i
*> 192.168.182.0/24 120.34.50.2      0         0 2762 1877 i

Displayed 6 routes and 10 total paths
frr#
```

**Gambar 3. 11 Verifikasi *routing BGP* pada *router central***

Verifikasi *BGP* menggunakan perintah *show ip bgp*, dari *router central* akan muncul ip dari *router* yang saling berkomunikasi dengan berbeda *AS*. Terdapat *IP* 10.100.10.10 yang merupakan *IP loopback* dari *router distributed*, artinya *router central* terhubung dengan *router distributed* dengan *next hop* melalui 2 jalur yaitu jalur *router agent 1* dengan *IP* 115.30.21.2 dan *router agent 2* dengan *IP* 120.34.50.2. Verifikasi tersebut juga menunjukkan *router central* terhubung dengan *router agent 1* dengan *IP loopback* 10.60.10.10 dan *router agent 2* dengan *IP loopback* 10.80.10.0. *Router central* juga terhubung dengan *network* dari *server* dengan *IP* 192.168.182.0.

```
frr# show ip bgp
BGP table version is 8, local router ID is 10.100.10.10, vrf id 0
Default local pref 100, local AS 1877
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
              i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @N.NN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network        Next Hop        Metric LocPrf Weight Path
*= 10.50.10.10/32  45.90.20.2      0         0 1427 1773 i
*> 10.50.10.10/32  38.22.120.2     0         0 2762 1773 i
*> 10.60.10.10/32  45.90.20.2      0         0 1427 i
*> 10.80.10.10/32  38.22.120.2     0         0 2762 i
*> 10.100.10.10/32 0.0.0.0          0         32768 i
*= 192.168.100.0/24 45.90.20.2      0         0 1427 1773 i
*> 192.168.100.0/24 38.22.120.2     0         0 2762 1773 i
*> 192.168.182.0/24 0.0.0.0          0         32768 i

Displayed 6 routes and 8 total paths
frr#
```

**Gambar 3. 12 Verifikasi *routing BGP* pada *router distributed***

Verifikasi *BGP* pada Gambar 3.12 *router distributed* menggunakan perintah *show ip bgp*, dari *router central* akan muncul *IP* dari *router* yang saling

berkomunikasi dengan berbeda AS. Terdapat *IP* 10.50.10.10 yang merupakan *IP loopback* dari *router central*, artinya *router central* terhubung dengan *router distributed* dengan *next hop* melalui 2 jalur yaitu jalur *router agent 1* dengan *IP* 45.90.20.2 dan *router agent 2* dengan *IP* 38.22.120.2. Verifikasi tersebut juga menunjukkan *router central* terhubung dengan *router agent 1* dengan *IP loopback* 10.60.10.10 dan *router agent 2* dengan *IP loopback* 10.80.10.0. *Router central* juga terhubung dengan *network* dari *client* dengan *IP* 192.168.100.0.

```
frr# show ip bgp
BGP table version is 17, local router ID is 10.60.10.10, vrf id 0
Default local pref 100, local AS 1427
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
              i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network        Next Hop        Metric LocPrf Weight Path
*> 10.50.10.10/32 115.30.21.1      0         0 1773 i
*> 10.60.10.10/32 0.0.0.0          0         32768 i
*= 10.80.10.10/32 115.30.21.1      0 1773 2762 i
*> 45.90.20.1      0 1877 2762 i
* 10.100.10.10/32 115.30.21.1      0 1773 2762 1877 i
*> 45.90.20.1      0 1877 i
*> 192.168.100.0/24 115.30.21.1      0 1773 i
* 192.168.182.0/24 115.30.21.1      0 1773 2762 1877 i
*> 45.90.20.1      0 1877 i

Displayed 6 routes and 9 total paths
frr#
```

**Gambar 3. 13 Verifikasi routing BGP pada Router Agent 1**

Verifikasi *BGP* pada Gambar 3/13 *router agent 1* menggunakan perintah *show ip bgp*, dari *router central* akan muncul *IP* dari *router* yang saling berkomunikasi dengan berbeda AS. Terdapat *IP* 10.80.10.10 yang merupakan *IP loopback* dari *router agent 2*, artinya *router agent 1* terhubung dengan *router agent 2*. *Router agent 1* juga terhubung dengan *router central* dengan *IP loopback* 10.100.10.10 dan terhubung dengan *router distributed* dengan *IP* 10.50.10.10. *Router agent 1* juga terhubung dengan *network* pada *client* dengan *IP* 192.168.100.0 dan *network* pada *server* dengan *IP* 192.168.182.0.

```
frr# show ip bgp
BGP table version is 13, local router ID is 10.80.10.10, vrf id 0
Default local pref 100, local AS 2762
Status codes: s suppressed, d damped, h history, * valid, > best, = multipath,
              i internal, r RIB-failure, S Stale, R Removed
Nexthop codes: @NNN nexthop's vrf id, < announce-nh-self
Origin codes: i - IGP, e - EGP, ? - incomplete

   Network        Next Hop        Metric LocPrf Weight Path
* 10.50.10.10/32 38.22.120.1      0 1877 1427 1773 i
*> 120.34.50.1      0 1773 i
*= 10.60.10.10/32 120.34.50.1      0 1773 1427 i
*> 38.22.120.1      0 1877 1427 i
*> 10.80.10.10/32 0.0.0.0          0 32768 i
*> 10.100.10.10/32 38.22.120.1      0 1877 i
* 192.168.100.0/24 38.22.120.1      0 1877 1427 1773 i
*> 120.34.50.1      0 1773 i
*> 192.168.182.0/24 38.22.120.1      0 1877 i

Displayed 6 routes and 9 total paths
frr#
```

**Gambar 3. 14 Verifikasi routing BGP pada Router Agent 2**

Verifikasi *BGP* pada Gambar 3.14 *router agent 2* menggunakan perintah *show ip bgp*, dari *router central* akan muncul *IP* dari *router* yang saling berkomunikasi dengan berbeda *AS*. Terdapat *IP* 10.60.10.10 yang merupakan *IP loopback* dari *router agent 1*, artinya *router agent 2* terhubung dengan *router agent 1*. *Router agent 1* juga terhubung dengan *router central* dengan *IP loopback* 10.100.10.10 dan terhubung dengan *router distributed* dengan *IP* 10.50.10.10. *Router agent 1* juga terhubung dengan *network* pada *client* dengan *IP* 192.168.100.0 dan *network* pada *server* dengan *IP* 192.168.182.0.

### 3.12 VERIFIKASI LOAD BALANCING

Setelah dilakukan konfigurasi *load balancing* dilakukan juga verifikasi apakah *load balancing* sudah tersedia pada jaringan yang diujikan.

```
frr# show ip route 10.100.10.10
Routing entry for 10.100.10.10/32
  Known via "bgp", distance 20, metric 0, best
  Last update 00:24:40 ago
    * 115.30.21.2, via eth0, weight 1
    * 120.34.50.2, via eth2, weight 1

frr# show ip bgp 10.100.10.10
BGP routing table entry for 10.100.10.10/32
Paths: (2 available, best #2, table default)
  Advertised to non peer-group peers:
    115.30.21.2 120.34.50.2
  1427 1877
    115.30.21.2 from 115.30.21.2 (10.60.10.10)
      Origin IGP, valid, external, multipath
      Last update: Wed Aug 23 23:24:02 2023
  2762 1877
    120.34.50.2 from 120.34.50.2 (10.80.10.10)
      Origin IGP, valid, external, multipath, best (Older Path)
      Last update: Wed Aug 23 23:24:01 2023
frr#
```

**Gambar 3. 15 Verifikasi *load balancing* pada *Router Central***

Verifikasi pada Gambar 3.15 pertama dilakukan dengan perintah *show ip route 10.00.10.10* yang mengarah ke *IP loopback* dari *router distributed*, hasilnya ada 2 jalur yang dilewati yaitu *router agent 1* dengan *IP* 115.39.21.2 dan *router agent 2* dengan *IP* 120.34.50.2 dengan *weight* yang sama yaitu 1. Verifikasi kedua dilakukan dengan perintah *show ip bgp 10.100.10.10* pada verifikasi tersebut menunjukkan ada 2 jalur *multipath* yaitu *router agent 1* dan 2 dengan *router agent 1* memiliki *ASN* 1427 dan *router agent 2* dengan *ASN* 2762, jalur tersebut bisa berperan sebagai jalur *request* dan jalur *reply*, konfigurasi *multipath* juga sudah aktif ketika dilakukan verifikasi.

```

frr# show ip route 10.50.10.10
Routing entry for 10.50.10.10/32
  Known via "bgp", distance 20, metric 0, best
  Last update 00:33:45 ago
    * 38.22.120.2, via eth3, weight 1
    * 45.90.20.2, via eth1, weight 1

frr# show ip bgp 10.50.10.10
BGP routing table entry for 10.50.10.10/32
Paths: (2 available, best #2, table default)
  Advertised to non-peer-group peers:
    38.22.120.2 45.90.20.2
    1427 1773
    45.90.20.2 from 45.90.20.2 (10.60.10.10)
      Origin IGP, valid, external, multipath
      Last update: Wed Aug 23 23:24:02 2023
    2762 1773
    38.22.120.2 from 38.22.120.2 (10.80.10.10)
      Origin IGP, valid, external, multipath, best (Older Path)
      Last update: Wed Aug 23 23:24:00 2023
frr#

```

**Gambar 3. 16 Verifikasi *load balancing* pada *router distributed***

Verifikasi pada Gambar 3.16 *load balancing* yang sama dilakukan dari sisi *router distributed*, perintah *show ip route 10.50.10.10* menunjukkan ada 2 jalur yang dilalui dari *router central ke router distributed* yaitu melalui *router agent 1* dengan IP 38.22.120.2 dan *router agent 2* dengan IP 45.90.20.2 dengan *weight* yang sama untuk kedua jalur. Verifikasi kedua dilakukan dengan perintah *show ip bgp 10.50.10.10* hasilnya ada 2 jalur *multipath* yaitu *router agent 1* dan *router agent 2*, dengan *router agent 1* memiliki ASN 1427 dan *router agent 2* dengan ASN 2762 dimana kedua jalur bisa berperan pada proses *request* dan *reply*, pada verifikasi tersebut terlihat konfigurasi *multipath* juga sudah aktif.

### 3.13 SKENARIO PENGUJIAN DATA

Skenario pengujian dilakukan dengan mengirim *file* dari *client* ke *server* dengan. Besaran data yang dikirimkan juga bervariasi mulai dari 100 MB, 256 MB, 512 MB dan 1 GB, dengan 5 pengujian pada masing masing besaran pengiriman data. Pengujian dilakukan pada topologi dengan konfigurasi *load balancing* dan topologi tanpa konfigurasi *load balancing*.

Dari hasil tersebut disimpulkan skenario pada Tabel 3.3 pengujian data di bawah ini bisa digunakan.

**Tabel 3. 3 Skenario Pengujian Data**

Besaran Data (MB)	Jumlah Pengujian
100	5

<b>Besaran Data (MB)</b>	<b>Jumlah Pengujian</b>
256	5
512	5
1000	5

Skenario pengujian dilakukan dengan menggunakan perintah *scp* yang digunakan untuk mengirim *file* dari *client* ke *server*, trafik pada saat pengiriman data akan di *capture* menggunakan *tools wireshark* dan dianalisa menggunakan perhitungan rumus *QoS throughput, delay* dan *packet loss*.