

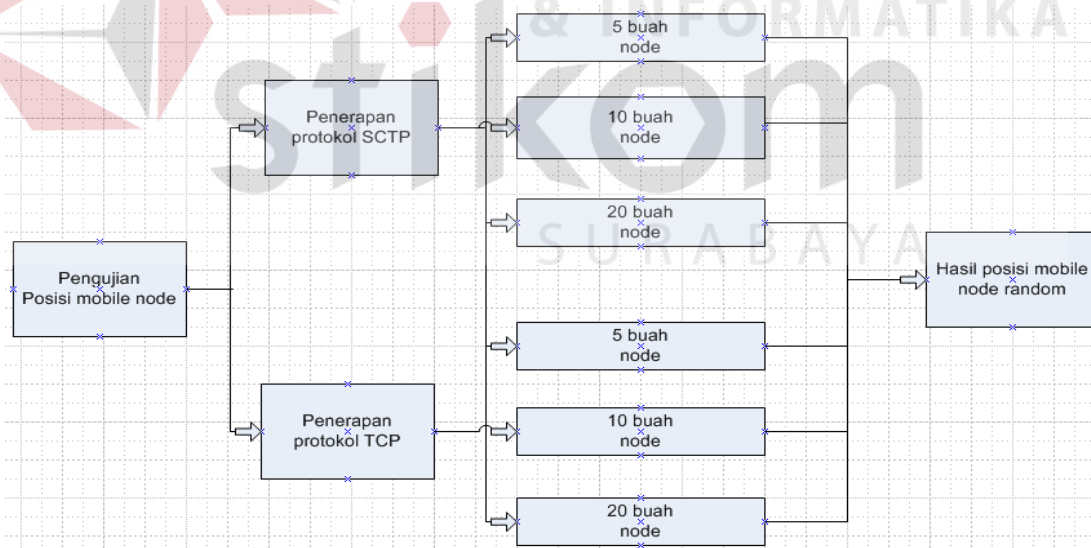
## BAB IV

### PENGUJIAN DAN EVALUASI

Pengujian sistem yang dilakukan merupakan pengujian terhadap simulasi yang telah selesai dibuat. Pengujian tersebut dimulai dari pengujian simulasi antara *mobile node*, pengujian protokol *routing*, dan pengujian terhadap parameter-parameter *QoS*, serta hasil analisis terhadap kedua protokol yang digunakan.

#### 4.1 Pengujian Posisi *Mobile Node*

Pengujian posisi *mobile node* dilakukan dengan menjalankan dari *script* “*random.tcl*” pada NS-2. Jika posisi *mobile node* berbeda-beda ketika menjalankan *script* yang satu dengan yang lain, maka topologi jaringan yang dibangkitkan secara *random* berhasil.



**Gambar 4.1** Blok Diagram Pengujian Posisi *Mobile Node*

Gambar 4.1 menjelaskan bahwa pengujian posisi *mobile node* dilakukan pada masing-masing penerapan protokol yaitu SCTP dan TCP baik pada 5 buah *node*, 10 buah *node*, dan 20 buah *node* secara terpisah. Pengujian tersebut digunakan untuk

melihat apakah *node-node mobile* yang dibangun pada simulasi berhasil mendapatkan posisi-posisi yang acak (*random*).

#### 4.1.1 Tujuan

Tujuan dari pengujian ini adalah untuk mengetahui apakah simulasi yang dibangun sudah berjalan sesuai dengan harapan yaitu membuat posisi *mobile node* secara *random*.

#### 4.1.2 Peralatan Yang Digunakan

##### 1. Perangkat keras (*Hardware*)

Hardware yang digunakan adalah sebuah Laptop dengan spesifikasi sebagai berikut :

- a. Processor : Intel(R) Core (TM) i3 CPU 2.13 GHz
- b. Sistem Operasi (SO) : Ubuntu 12.04 LTS
- c. Memory : 4 GB

##### 2. Perangkat Lunak (*Software*)

- a. NS-2 2.34 : adalah aplikasi utama yang digunakan untuk membuat *script* Tcl.
- b. Microsoft excel 2007 : adalah *software* yang digunakan untuk melakukan *filter* data serta perhitungan.

#### 4.1.3 Prosedur Pengujian *Mobile Node*

Langkah-langkah untuk melakukan pengujian ini adalah sebagai berikut :

1. Nyalakan laptop dengan SO Ubuntu 12.04.
2. Buka jendela terminal pada menu bar.

3. Kemudian akses ke direktori tempat *script* tersebut disimpan. Dalam penelitian ini, direktori yang digunakan untuk menyimpan *script* tersebut adalah pada direktori “home/jojo/desktop/Tugasakhir” secara keseluruhan. Sedangkan untuk *script* dengan menggunakan algoritma TCP dan SCTP terpisah di dalam direktori Tugas akhir tersebut yaitu direktori “TCP” dan “SCTP”. Kemudian untuk *script* yang dibangun menggunakan 5 buah *node*, 10 buah *node*, dan 20 *node* dibagi menjadi direktori pada masing-masing direktori “TCP” dan “SCTP”. Sebagai contoh, untuk mengakses *script* yang menggunakan algoritma TCP 5 buah *node* pada percobaan pertama dengan perintah sebagai berikut :

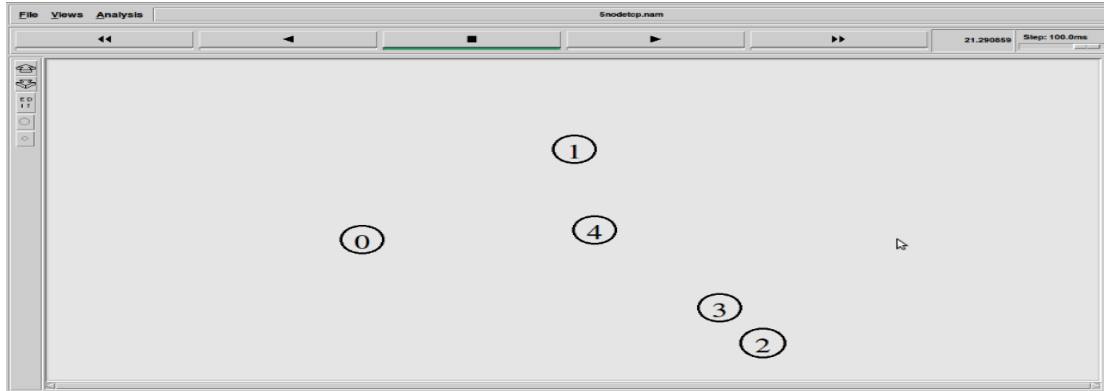
```
root@ubuntu: /home/jojo/desktop/TA5node
```

4. Setelah berhasil mengakses direktori yang digunakan di atas, *script* “*random.tcl*” dapat dijalankan dengan perintah :

```
root@ubuntu: /home/jojo/desktop/ns TA5node.tcl
```

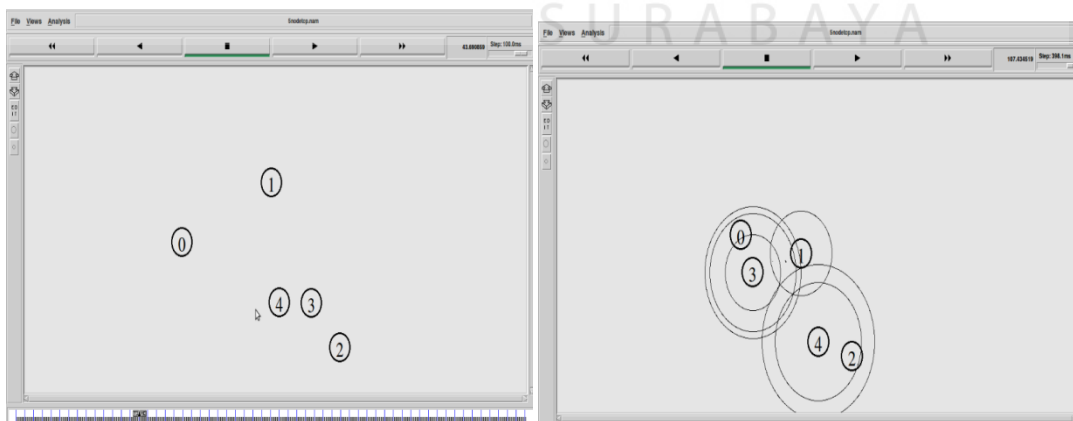
#### 4.1.4 Hasil Pengujian Posisi *Mobile Node*

Pengujian posisi *mobile node* ini digunakan untuk memastikan bahwa posisi dari *node-node* ketika *script* dijalankan bersifat *random*. Posisi-posisi yang dihasilkan dapat dilihat pada jendela *Network Animator* (NAM). Berikut adalah beberapa hasil dari menjalankan *script* “TA5node.tcl” untuk masing-masing skenario :



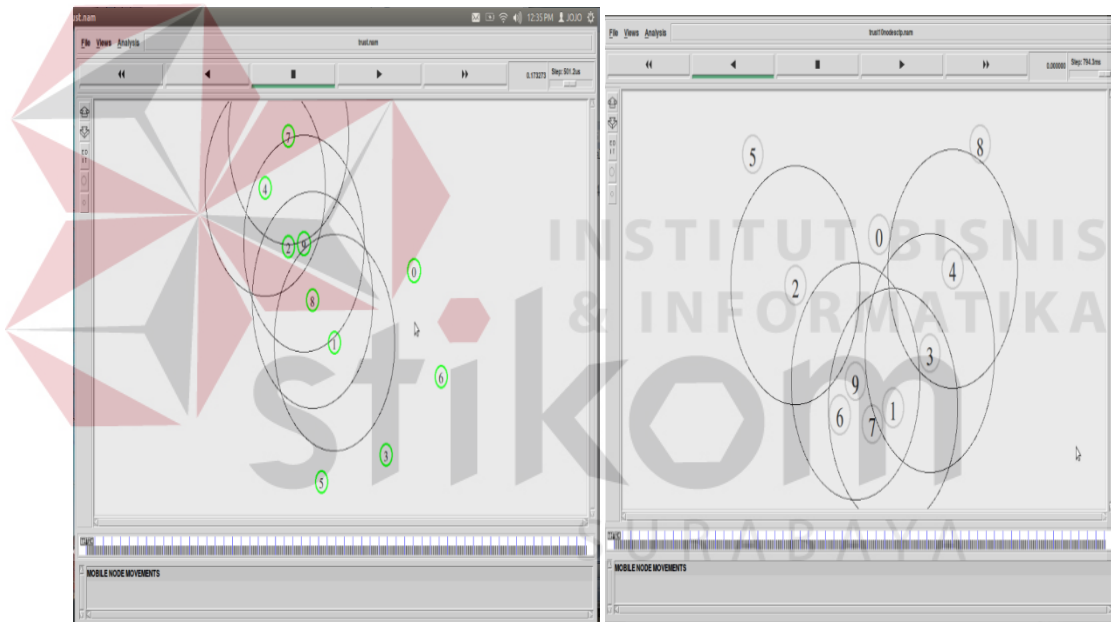
**Gambar 4.2** Hasil NAM Penerapan Protokol TCP Pada 5 Buah *Mobile Node* Detik Ke-21

Pada Gambar 4.2 dapat dilihat bahwa pemanggilan terhadap *script* "TA5node.tcl" menghasilkan topologi dengan posisi-posisi *mobile node* yang berbeda-beda pada percobaan di setiap detiknya. Hal tersebut menunjukkan bahwa topologi yang dibangun secara *random* berhasil dibangun dimana letak posisi *node* 0-4 pada percobaan pertama di detik ke-21 berbeda dengan letak posisi *node* 0-4 pada detik berikutnya. Hal tersebut juga terjadi pada percobaan-percobaan lain baik yang menggunakan 5 buah *node*, 10 buah *node*, maupun 20 buah *node* dengan protokol yang berbeda yang akan ditunjukkan pada Gambar 4.3 s/d Gambar 4.5.



**Gambar 4.3** Hasil NAM Penerapan Protokol TCP Pada 5 Buah *Mobile Node* Detik Ke-43 (kiri) Dan Detik Ke-107 (kanan)

Pada Gambar 4.3 dapat dilihat bahwa pemanggilan terhadap “TA5node.tcl” menghasilkan topologi dengan posisi-posisi *mobile node* yang berbeda antara percobaan pada detik ke-43 dan detik ke-107. Hal tersebut menunjukkan bahwa topologi yang dibangun secara *random* berhasil dibangun dimana letak posisi *node* 0-4 pada percobaan pertama berbeda-beda di setiap detik selanjutnya. Hal tersebut juga terjadi pada percobaan-percobaan lain baik yang menggunakan 5 buah *node* 10 buah *node*, dan 20 buah *node* dengan protokol yang berbeda yang akan di tunjukkan pada Gambar 4.4 dan Gambar 4.5.



**Gambar 4.4** Hasil NAM Penerapan Protokol SCTP Pada 10 Buah *Mobile Node*

Detik Ke-10 (kiri) Dan Detik Ke-60 (kanan)

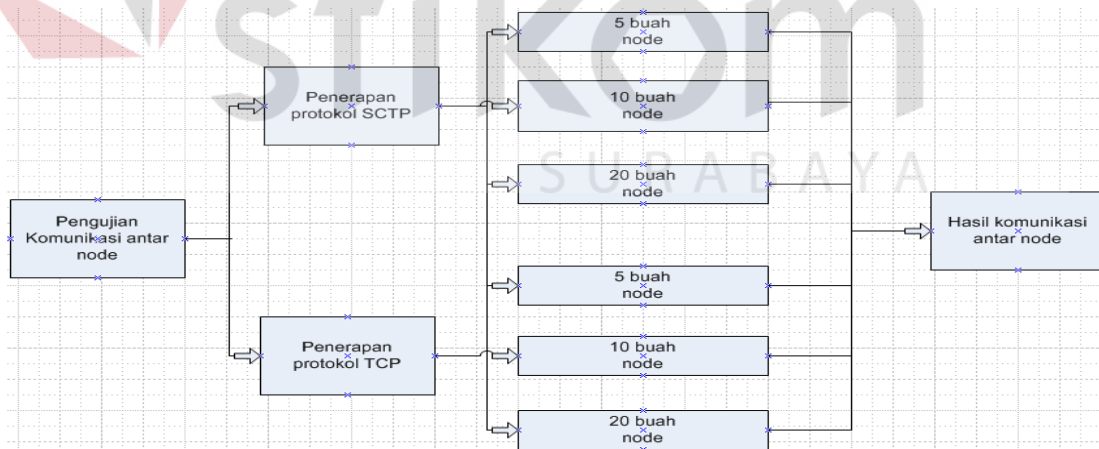


**Gambar 4.5** Hasil NAM Penerapan Protokol SCTP Pada 20 Buah *Mobile Node*

Detik Ke-30 (kiri) Dan Detik Ke-90 (kanan)

#### 4.2. Pengujian Komuniaksi Antar *Mobile Node*

Pengujian komunikasi antar *mobile node* dilakukan dengan menjalankan NAM dari script “TA5node.tcl” pada NS-2. Jika *node* sumber berhasil mengirimkan data menuju *node* penerima, maka dapat diketahui bahwa komunikasi antar *node* berhasil.



**Gambar 4.6** Blok Diagram Pengujian Komunikasi Antar *Mobile Node*

Gambar 4.6 menjelaskan bahwa pengujian komunikasi antar *mobile node* dilakukan pada masing-masing penerapan protokol yaitu TCP dan SCTP baik pada 5 buah *node*, 10 buah *node*, dan 20 buah *node* secara terpisah. Pengujian tersebut

digunakan untuk melihat apakah *node-node* yang terlibat dalam komunikasi pengiriman data pada simulasi dapat saling berkomunikasi baik mengirim sinyal maupun mengirim dan menerima data.

#### 4.2.1 Tujuan

Tujuan dari pengujian ini adalah untuk mengetahui apakah simulasi yang dibangun sudah berjalan sesuai dengan harapan. Serta untuk menguji apakah masing-masing *mobile node* dapat melakukan komunikasi.

#### 4.2.2 Peralatan Yang Digunakan

##### 1. Perangkat keras (*Hardware*)

Hardware yang digunakan adalah sebuah Laptop dengan spesifikasi sebagai berikut :

- a. Processor : Intel(R) Core (TM) i3 CPU 2.13 GHz
- b. Sistem Operasi (SO) : Ubuntu 12.04 LTS
- c. Memory : 4 GB

##### 2. Perangkat Lunak (*Software*)

- 2.1 NS-2 2.34 : adalah aplikasi utama yang digunakan untuk membuat *script* Tcl.
- 2.2 Microsoft Excel 2007 : digunakan untuk mengolah data-data yang telah difilter.

#### 4.2.3 Prosedur Pengujian Komunikasi *Mobile Node*

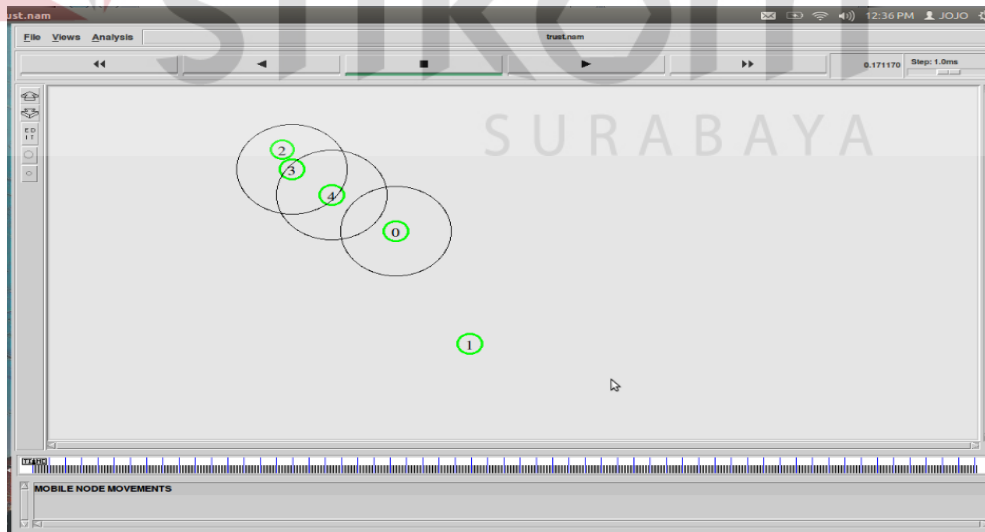
Langkah-langkah untuk melakukan pengujian ini adalah sebagai berikut :

1. Dari jendela NAM yang dihasilkan pada pengujian sebelumnya, NAM tersebut harus dijalankan yaitu dengan cara menekan tombol “start (▶)” yang berada dalam menu bar jendela NAM sampai dengan akhir dari simulasi selama 180 detik.
2. Menunggu hingga simulasi berakhir.

#### 4.2.4 Hasil Pengujian Komuniaksi Antar *Mobile Node*

Komunikasi antar *mobile node* berlangsung ketika simulasi berjalan. Komunikasi tersebut dimulai dari saling bertukar informasi antar *node* ditandai dengan adanya sinyal-sinyal yang dipancarkan oleh *node-node* yang berdekatan yang mampu dijangkau oleh setiap *node* yang bersangkutan serta proses pengiriman paket oleh *node* sumber menuju *node* tujuan. Keberhasilan komunikasi tersebut dapat dilihat pada hasil berikut :


1. Penerapan protokol TCP pada 5 buah *mobile node*



**Gambar 4.7** Hasil Komunikasi 5 Buah *Mobile Node* Dengan Protokol TCP

Pada Gambar 4.7 dapat dilihat bahwa *node* sumber adalah *node* 0 dan *node* tujuan adalah *node* 4. Proses saling menukar informasi antar *node* dapat dilihat




dengan sinyal lingkaran dari *node* 0, *node* 3, dan *node* 4, hal tersebut menunjukkan bahwa *node* yang dapat di jangkau oleh *node* sumber yaitu *node* 3 dan *node* 4. Karena posisi *node* 4 tidak terlalu jauh dari *node* sumber, maka pengiriman data dilakukan tanpa perantara *node* lain. Proses pengiriman data dapat dilihat pada gambar “” pada Gambar 4.7. Sehingga dapat dikatakan bahwa komunikasi antar *node* berhasil pada percobaan penerapan protokol TCP pada 5 buah *node*.

2. Penerapan protokol TCP pada 10 buah *mobile node*.

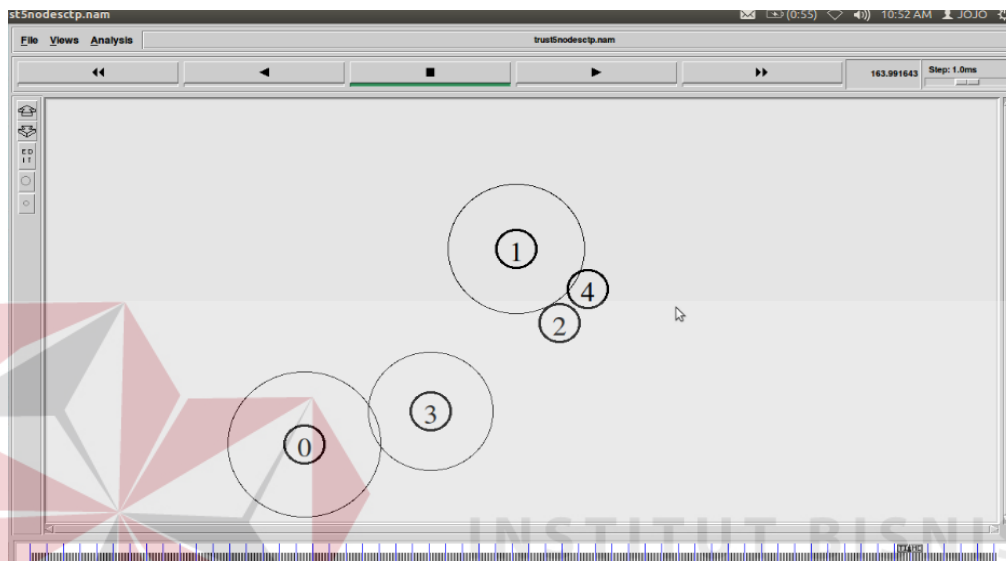


**Gambar 4.8** Hasil Komunikasi 10 Buah *Mobile Node* Dengan Protokol TCP


Pada Gambar 4.8 dapat dilihat bahwa *node* sumber adalah *node* 1 dan *node* tujuan adalah *node* 2. Proses saling menukar informasi antar *node* dapat dilihat dengan sinyal lingkaran dari *node* 1, *node* 2, *node* 3, *node* 4, *node* 9, dan sebuah lingkaran yang melingkupi *node* 6 dan *node* 7, hal tersebut menunjukkan bahwa *node* yang dapat dijangkau oleh *node* sumber yaitu *node* 2, *node* 3, *node* 4, *node* 6, *node* 7, dan *node* 9. Karena posisi *node* 1 tidak terlalu jauh dari *node* sumber, maka pengiriman data dilakukan tanpa perantara *node* lain. Proses pengiriman data dapat

dilihat pada gambar “” pada Gambar 4.8. Sehingga dapat dikatakan bahwa komunikasi antar *node* berhasil pada percobaan penerapan protokol TCP pada 10 buah *node*.

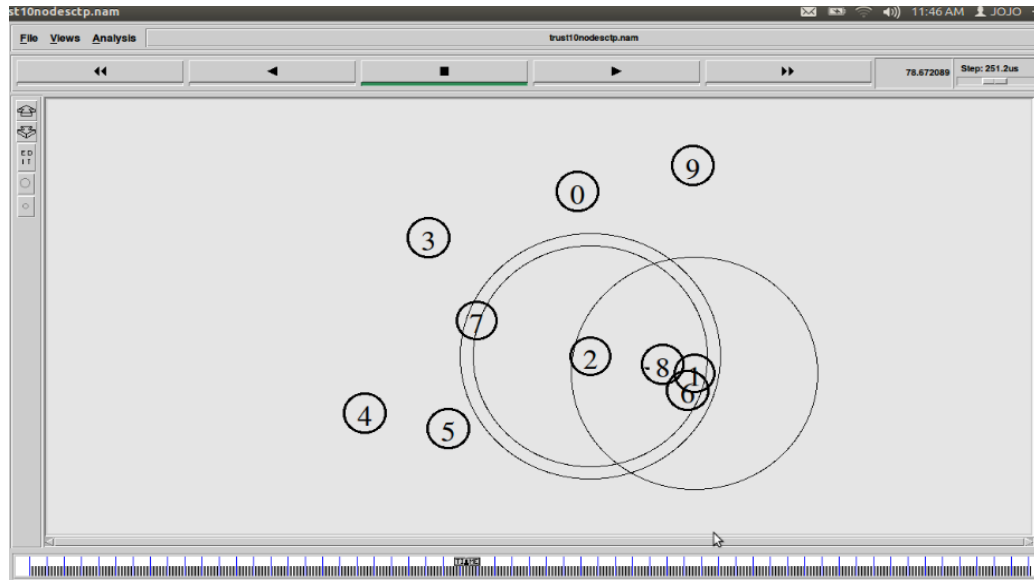
### 3. Penerapan protokol SCTP pada 5 buah *mobile node*




**Gambar 4.9** Hasil Komunikasi 5 Buah *Mobile Node* Dengan Protokol SCTP

Pada Gambar 4.9 dapat dilihat bahwa *node* sumber adalah *node* 1 dan *node* tujuan adalah *node* 4. Proses saling menukar informasi antar *node* dapat dilihat dengan sinyal lingkaran pada *node* 1 dan *node* 4, dan 2 lingkaran yang terlihat pada *node* 0 dan *node* 3. Hal tersebut menunjukkan bahwa *node* yang dapat dijangkau oleh *node* sumber yaitu *node* 2, *node* 0, *node* 3, dan *node* 4. Karena posisi *node* 4 tidak terlalu jauh dari *node* sumber, maka pengiriman data dilakukan tanpa perantara *node* lain. Proses pengiriman data dapat dilihat pada gambar “” pada Gambar 4.9. Sehingga dapat dikatakan bahwa komunikasi antar *node* berhasil pada percobaan penerapan protokol SCTP pada 5 buah *node*.

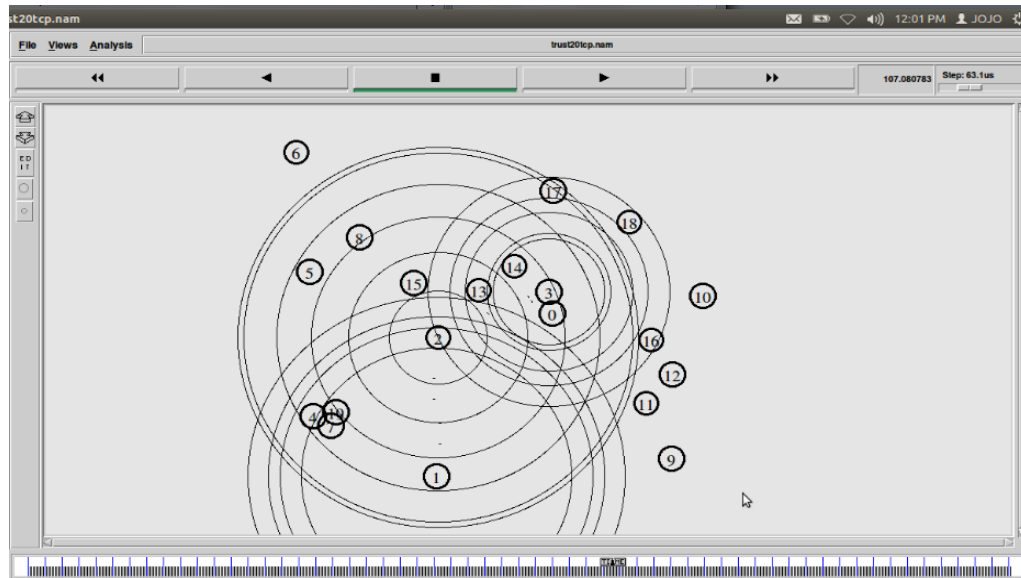
### 4. Penerapan protokol SCTP pada 10 buah *mobile node*




**Gambar 4.10** Hasil Komunikasi 10 Buah *Mobile Node* Dengan Protokol SCTP

Pada Gambar 4.10 dapat dilihat bahwa *node* sumber adalah *node* 1 dan *node* tujuan adalah *node* 2. Proses saling menukar informasi antar *node* dapat dilihat dengan sinyal lingkaran pada *node* 1, dan *node* 2, serta sebuah lingkaran yang melingkupi *node* 7, *node* 8, dan *node* 6. Hal tersebut menunjukkan bahwa *node* yang dapat dijangkau oleh *node* sumber yaitu *node* 2, *node* 6, *node* 7, dan *node* 8. Karena posisi *node* 2 tidak terlalu jauh dari *node* sumber, maka pengiriman data dilakukan tanpa perantara *node* lain. Proses pengiriman data dapat dilihat pada gambar “” pada Gambar 4.10. Sehingga dapat dikatakan bahwa komunikasi antar *node* berhasil pada percobaan penerapan protokol SCTP pada 10 buah *node*.

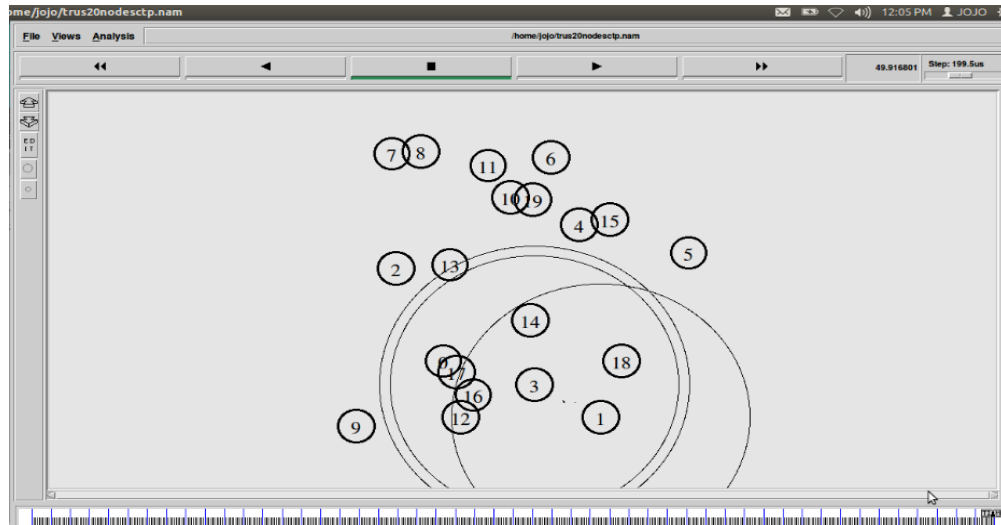
5. Penerapan protokol TCP pada 20 buah *mobile node*




**Gambar 4.11** Hasil Komunikasi 20 Buah *Mobile Node* Dengan Protokol TCP

Pada Gambar 4.11 dapat dilihat bahwa *node* sumber adalah *node* 1 dan *node* tujuan adalah *node* 3. Proses saling menukar informasi antar *node* dapat dilihat dengan sinyal lingkaran pada *node* 1, dan *node* 3, serta 3 buah lingkaran yang melingkupi *node* 4, *node* 5, *node* 7, *node* 8, *node* 13, *node* 14, *node* 15, *node* 16, *node* 17, *node* 18 dan *node* 19. Hal tersebut menunjukkan bahwa *node* yang dapat dijangkau oleh *node* sumber yaitu *node* 3, *node* 4, *node* 5, *node* 7, *node* 8, *node* 13, *node* 14, *node* 15, *node* 16, *node* 17, *node* 18 dan *node* 19. Karena posisi *node* 3 terlalu jauh dari *node* sumber, maka pengiriman data dilakukan lewat perantara *node* 2 dan di teruskan ke *node* 3 sebagai tujuannya. Proses pengiriman data dapat dilihat pada gambar “” pada Gambar 4.11. Sehingga dapat dikatakan bahwa komunikasi antar *node* berhasil pada percobaan penerapan protokol TCP pada 20 buah *node*.

#### 6. Penerapan protokol SCTP pada 20 buah *mobile node*

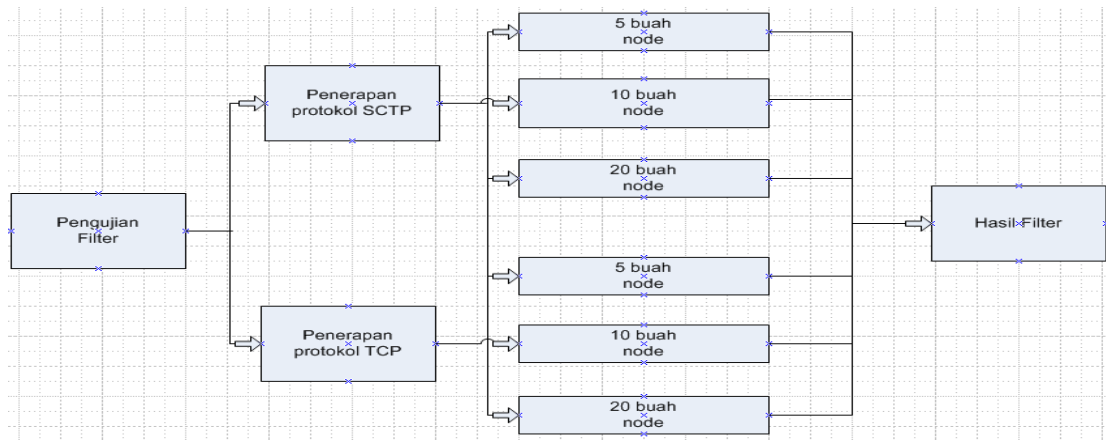


**Gambar 4.12** Hasil Komunikasi 20 Buah *Mobile Node* Dengan Protokol SCTP

Pada Gambar 4.12 dapat dilihat bahwa *node* sumber adalah *node* 1 dan *node* tujuan adalah *node* 3. Proses saling menukar informasi antar *node* dapat dilihat dengan sinyal lingkaran pada *node* 1, dan *node* 3, serta sebuah lingkaran yang melingkupi *node* 0, *node* 12, *node* 13, *node* 14, *node* 16, *node* 17, dan *node* 18. Hal tersebut menunjukkan bahwa *node* yang dapat dijangkau oleh *node* sumber yaitu *node* 3, *node* 0, *node* 12, *node* 13, *node* 14, *node* 16, *node* 17, dan *node* 18. Karena posisi *node* 3 tidak terlalu jauh dari *node* sumber, maka pengiriman data dilakukan secara langsung tanpa perantara *node* lain. Proses pengiriman data dapat dilihat pada gambar “” pada Gambar 4.12. Sehingga dapat dikatakan bahwa komunikasi antar *node* berhasil pada percobaan penerapan protokol SCTP pada 20 buah *node*.

### 4.3. Pengujian *Filter*

Pengujian *filter* dilakukan dengan cara menjalankan script “*filter.pl*” pada jendela terminal. Sehingga dapat dihasilkan data-data yang telah dipilah berdasarkan kebutuhan pengolahan data.



**Gambar 4.13** Blok Diagram Pengujian *Filter*

Gambar 4.13 menjelaskan bahwa pengujian *filter* data dilakukan pada masing-masing hasil penerapan protokol yaitu TCP dan SCTP baik pada 5 buah *node*, 10 buah *node* dan, 20 buah *node* secara terpisah. Pengujian tersebut digunakan untuk melihat apakah data-data yang dibutuhkan berhasil difilter.

### 4.3.1 Tujuan

Tujuan dari pengujian ini adalah untuk mengetahui apakah data-data yang berada pada *tracefile* hasil simulasi berhasil disaring sesuai dengan kebutuhan pengolahan data.

### 4.3.2 Peralatan Yang Digunakan

#### 1. Perangkat keras (*Hardware*)

Hardware yang digunakan adalah sebuah Laptop dengan spesifikasi sebagai berikut :

- a. Processor : Intel(R) Core (TM) i3 CPU 2.13 GHz
- b. Sistem Operasi (SO) : Ubuntu 12.04 LTS
- c. Memory : 4 GB

## 2. Perangkat Lunak (*Software*)

- a. NS-2 2.34 : adalah aplikasi utama yang digunakan untuk membuat *script* Tcl.
- b. Microsoft Excel 2007 : digunakan untuk mengolah data-data yang telah difilter.

### 4.3.3 Prosedur Pengujian

Langkah-langkah untuk melakukan pengujian ini adalah sebagai berikut :

1. Buka jendela terminal pada menu bar.
2. Masuk ke sistem “*root*” administrator sebelum menjalankan script “*filter.pl*”. Dengan menggunakan perintah :

```
jojo@ubuntu: ~$ sudo su
```

3. Kemudian akses ke direktori tempat script tersebut disimpan. Dalam penelitian ini, direktori yang digunakan untuk menyimpan script tersebut adalah pada direktori “*home/jojo/Desktop/TA5node.tcl*” yang dibagi menjadi 2 direktori yaitu direktori yang menunjukkan penerapan protokol TCP dan SCTP yaitu direktori “TCP” dan “SCTP”. Kemudian dibagi lagi berdasarkan *node* yang digunakan yaitu 5 buah *node*, 10 buah *node*, dan 20 buah *node*. Sebagai contoh, untuk mengakses script yang menggunakan protokol TCP dengan 5 buah *node* pada percobaan pertama dengan perintah sebagai berikut :

```
root@ubuntu: /home/jojo/Desktop/TCP5node
```

4. Setelah berhasil mengakses direktori yang digunakan di atas, script “*filter.pl*” dapat dijalankan dengan perintah :

```
root@ubuntu: /home/jojo/Desktop/TCP5node#
perl filtr.pl simple.tr > filter
```

Perintah tersebut menunjukkan bahwa script “*filter.pl*” dijalankan untuk memfilter data-data yang tersimpan pada file “*simple.tr*” yang kemudian hasil filter tersebut disimpan ke dalam file baru dengan nama “*filter*”.

#### 4.3.4 Hasil Pengujian *Filter*

Data yang dibutuhkan hanya data dengan tipe paket tcp dan sctp saja. Informasi-informasi yang dibutuhkan adalah informasi pada kolom 0, kolom 1, kolom 2, kolom 3, kolom 5, kolom 6, kolom 7, kolom 8, dan kolom 9. Sehingga menghasilkan *file filter* seperti Gambar 4.14

1 s		10.0.0.0	40	-1	[0]
2 AGT	5	10.0.0.0	40	-1	[0]
3	0	10.0.0.0	40	-1	[0]
4 r	5	13.0.0.0	40	-1	[0]
5 RTR	12	13.0.0.0	40	-1	[0]
6	0	13.0.0.0	40	-1	[0]
7 s	12	19.0.0.0	40	-1	[0]
8 AGT	23	19.0.0.0	40	-1	[0]
9	0	19.0.0.0	40	-1	[0]
10 r	12	19.0.0.0	40	-1	[0]
11 RTR	23	19.0.0.0	40	-1	[0]
12	0	19.0.0.0	40	-1	[0]
13 s	23	27.523527564	80	-1	[0]
14 AGT	5	27.523527564	80	-1	[0]
15	0	27.523527564	80	-1	[0]
16 r	12	27.523527564	80	-1	[0]
17 RTR	23	27.523527564	80	-1	[0]
18	0	27.523527564	80	-1	[0]
19 s	5	27.523527564	80	-1	[0]
20 RTR	12	27.523527564	80	-1	[0]
21	0	27.523527564	80	-1	[0]
22 s	12	27.523527564	80	-1	[0]
23 RTR	23	27.523527564	80	-1	[0]
24	0	27.523527564	80	-1	[0]
25 D	5	27.523527564	80	-1	[0]
26 IFQ	1	27.523527564	80	-1	[0]
27	23	27.523527564	80	-1	[0]
28 s	0	27.523527564	80	-1	[0]
29 RTR	12	27.523527564	80	-1	[0]
30	1	27.523527564	80	-1	[0]
31 D	23	27.530933167	138	-1	[13a]
32 IFQ	1	27.530933167	80	-1	[0]
33	23	27.532037627	80	-0	[13a]
34 s					
35 MAC					
36					
37 r					
38 MAC					

**Gambar 4.14** Hasil *Filter “simple.tr”* Penerapan Protokol TCP Pada 5 Buah *Node*



Hasil yang terlihat pada Gambar 4.14 yang akan digunakan untuk diolah, sehingga menghasilkan *delay*, *utilisasi bandwidth*, dan paket *loss*. Kemudian akan diplot ke dalam bentuk grafik sehingga dapat dilihat perbandingannya.

#### 4.4. Hasil Perhitungan Parameter-Parameter *Qos*

##### 4.4.1 Hasil Perhitungan *Delay*

Dari data-data yang sudah tersimpan dalam file “*filter*” hasil dari *filter* dapat diperoleh *delay* dari masing-masing percobaan. *Delay* tersebut dapat diperoleh dengan menggunakan rumus penghitungan *delay* sebagai berikut :

$$\text{Waktu tunda } (t) = (Tr - Ts) \text{ detik} \dots\dots\dots(4.1)$$

Dimana :  $0 \leq t \leq T$

Dengan :

$T_r$  = Waktu penerimaan paket (detik)

$T_s$  = Waktu pengiriman paket (detik)

$T$  = Waktu simulasi (detik)

$t$  = Waktu pengambilan sampel (detik)

Delay yang terjadi pada masing-masing percobaan berbeda-beda.

Adapun hasil perhitungan *delay* dapat dilihat pada Tabel 4.1.

##### 4.4.1.1 Perhitungan *Delay* 5 Buah *Node* TCP Dan SCTP

**Tabel 4.1** Hasil Penghitungan *Delay* Protokol TCP Dan SCTP Pada 5 Buah *Node*

Percobaan Ke-	TCP(s)	SCTP(s)
1	0.00491125	0.009646092
2	0.009535355	0.00666695
3	0.005198015	0.030002884

Percobaan Ke-	TCP(s)	SCTP(s)
4	0.007787407	0.028835336
5	0.00969999	0.012144952
6	0.008088928	0.006666851
7	0.006703637	0.019163473
8	0.010111162	0.005130688
9	0.002857144	0.008541635
10	0.006667	0.004010042

Dari Tabel 4.1 dapat dilihat bahwa rata-rata *delay* yang terjadi pada penerapan protokol TCP dan SCTP pada 5 buah *node* yang tersebar secara *random* adalah :

$$\text{Rata - rata delay tcp} = (\sum \text{delay}/10) = (0.071559889/10) = 0.007155989 \text{ detik}$$

Dengan rumus standar deviasi: (Walpole, Ronald E., Raymond H Myers, 1995).

$$S = \sqrt{\frac{\sum (X_i - (\text{rata-rata delay}))^2}{(n-1)}} = 0.0023554133930917 \text{ detik}$$

Interval rata-rata *delay* tcp dihitung dengan cara:

$$\begin{aligned} \text{Interval} &= 0.007155989 \pm Z_{0.05/2} \cdot (0.0023554133930917 / \sqrt{10}) \\ &= 0.007155989 \pm Z_{0.025} \cdot (0.0023554133930917 / \sqrt{10}) \\ &= 0.007155989 \pm 1.96 \cdot (0.000744847) \\ &= 0.007155989 \pm 0.0014599012 \\ &= (0.007155989 - 0.0014599012 ; 0.007155989 + 0.0014599012) \\ &= (0.00569608888 ; 0.00861588912) \end{aligned}$$

Jadi, nilai *delay* yang dihasilkan pada penerapan TCP dengan 5 buah *node* adalah sekitar 0.00569608888 detik s/d 0.00861588912 detik.

Sedangkan untuk SCTP :

*Rata – rata delay sctp* =  $(\sum \text{delay}/10) = (0.130808903/10) = 0.01308089$  detik

$$S = \sqrt{\frac{\sum (xi - (\text{rata-rata delay}))^2}{(n-1)}} = 0.009618254847179 \text{ detik}$$

Interval rata-rata *delay sctp* dihitung dengan cara:

$$\text{Interval} = 0.01308089 \pm Z_{0.05/2} \cdot (0.009618254847179 / \sqrt{10})$$

$$= 0.01308089 \pm Z_{0.025} \cdot (0.009618254847179 / \sqrt{10})$$

$$= 0.01308089 \pm 1.96 \cdot (0.00304155)$$

$$= 0.01308089 \pm 0.00596145611$$

$$= (0.01308089 - 0.00596145611 ; 0.01308089 + 0.00596145611)$$

$$= (0.00711943388 ; 0.01904234611)$$

Jadi, nilai *delay* yang dihasilkan pada penerapan SCTP dengan 5 buah *node* adalah sekitar 0.00711943388 detik s/d 0.01904234611 detik.

#### 4.4.1.2 Perhitungan Delay 10 Buah Node TCP Dan SCTP

**Tabel 4.2** Hasil Penghitungan *Delay* Protokol TCP Dan SCTP Pada 10 Buah *Node*

Percobaan Ke-	TCP(s)	SCTP(s)
1	0.006	0.001962013
2	0.007500003	0.004955863
3	0.018000001	0.00666295
4	0.005244642	0.005115464

Percobaan Ke-	TCP(s)	SCTP(s)
5	0.0007	0.005087
6	0.004166667	0.009745567
7	0.0019	0.002232223
8	0.00775	0.003055375
9	0.008363636	0.001753685
10	0.009648333	0.001878295

Dari Tabel 4.2 dapat dilihat bahwa rata-rata delay yang terjadi pada penerapan protokol TCP 10 buah *node* yang tersebar secara random adalah :

$$\text{Rata - rata delay TCP} = (\sum \text{delay}/10) = (0.069273282/10) = 0.006927328 \text{ detik}$$

$$S = \sqrt{\frac{\sum (x_i - (\text{rata-rata delay}))^2}{(n-1)}} = 0.0048125061790586 \text{ detik}$$

Interval rata-rata *delay* tcp dihitung dengan cara:

$$\begin{aligned} \text{Interval} &= 0.006927328 \pm Z_{0.05/2} \cdot (0.0048125061790586 / \sqrt{10}) \\ &= 0.006927328 \pm Z_{0.025} \cdot (0.0048125061790586 / \sqrt{10}) \\ &= 0.006927328 \pm 1.96 \cdot (0.001521848) \\ &= 0.006927328 \pm 0.002982822 \\ &= (0.006927328 - 0.002982822; 0.006927328 + 0.002982822) \\ &= (0.003944506 ; 0.00991015) \end{aligned}$$

Jadi, nilai *delay* yang dihasilkan pada penerapan TCP dengan 10 buah *node* adalah sekitar 0.003944506 detik s/d 0.00991015 detik.

Sedangkan untuk SCTP :

$Rata - rata\ delay = (\sum delay/10) = (0.042448435/10) = 0.004244844\ detik$

$$S = \sqrt{\frac{\sum (xi - (rata-rata\ delay))^2}{(n-1)}} = 0.0025940352216384\ detik$$

Interval rata-rata *delay* sctp dihitung dengan cara :

$$\begin{aligned} \text{Interval} &= 0.004244844 \pm Z_{0.05/2} \cdot (0.0025940352216384 / \sqrt{10}) \\ &= 0.004244844 \pm Z_{0.025} \cdot (0.0025940352216384 / \sqrt{10}) \\ &= 0.004244844 \pm 1.96 \cdot (0.000820305) \\ &= 0.004244844 \pm 0.0016077978 \\ &= 0.004244844 - 0.0016077978 ; 0.004244844 + 0.0016077978 \\ &= (0.0026370462 ; 0.0058526418) \end{aligned}$$

Jadi, nilai *delay* yang dihasilkan pada penerapan SCTP dengan 10 buah *node* adalah sekitar 0.0026370462 detik s/d 0.0058526418 detik.

#### 4.4.1.3 Perhitungan Delay 20 Buah Node TCP Dan SCTP

**Tabel 4.3** Hasil Penghitungan Delay Protokol TCP Dan SCTP Pada 20 Buah *Node*

Percobaan Ke-	TCP(s)	SCTP(s)
1	0.0085001	0.010033358
2	0.009287617	0.016833336
3	0.010000031	0.007575002
4	0.00775018	0.000464013
5	0.005525007	0.000540011
6	0.007951746	0.000500017
7	0.008500001	0.011250002

Percobaan Ke-	TCP(s)	SCTP(s)
8	0.007750009	0.003400001
9	0.008500008	0.026200007
10	0.007750032	0.00802222

Dari Tabel 4.3 dapat dilihat bahwa rata-rata delay yang terjadi pada penerapan protokol TCP 20 buah *node* yang tersebar secara *random* adalah :

$$\text{Rata - rata delay} = (\sum \text{delay}/10) = (0.081514731/10) = 0.008151473 \text{ detik}$$

$$S = \sqrt{\frac{\sum (x_i - (\text{rata-rata delay}))^2}{(n-1)}} = 0.0011782311545301 \text{ detik}$$

Interval rata-rata *delay* tcp dihitung dengan cara:

$$\begin{aligned} \text{Interval} &= 0.008151473 \pm Z_{0.05/2} \cdot (0.0011782311545301 / \sqrt{10}) \\ &= 0.008151473 \pm Z_{0.025} \cdot (0.0011782311545301 / \sqrt{10}) \\ &= 0.008151473 \pm 1.96 \cdot (0.000372589) \\ &= 0.008151473 \pm 0.00073027444 \\ &= (0.008151473 - 0.00073027444 ; 0.008151473 + 0.00073027444) \\ &= 0.00742119856 ; 0.00888174744) \end{aligned}$$

Jadi, nilai *delay* yang dihasilkan pada penerapan TCP dengan 20 buah *node* adalah sekitar 0.00742119856 detik s/d 0.00888174744detik.

Sedangkan untuk SCTP :

$$\text{Rata - rata delay} = (\sum \text{delay}/10) = (0.084817967/10) = 0.008481797 \text{ detik}$$

$$S = \sqrt{\frac{\sum (x_i - (\text{rata-rata delay}))^2}{(n-1)}} = 0.0082440365356945 \text{ detik}$$

Interval rata-rata *delay* sctp dihitung dengan cara:

$$\begin{aligned} \text{Interval} &= 0.008481797 \pm Z_{0.05/2} \cdot (0.0082440365356945 / \sqrt{10}) \\ &= 0.008481797 \pm Z_{0.025} \cdot (0.0082440365356945 / \sqrt{10}) \\ &= 0.008481797 \pm 1.96 \cdot (0.00260699) \\ &= 0.008481797 \pm 0.005109700 \\ &= (0.008481797 - 0.005109700; 0.008481797 + 0.005109700) \\ &= (0.003372097; 0.013591497) \end{aligned}$$

Jadi, nilai *delay* yang dihasilkan pada penerapan SCTP dengan 20 buah *node* adalah sekitar 0.003372097 detik s/d 0.013591497detik.

#### 4.4.2 Hasil Perhitungan PLR (*Paket Loos Ratio*)

Sama seperti perhitungan yang sebelumnya, PLR dihitung dari *file* “*filter*” yang sudah ada sebelumnya. Adapun cara untuk memperoleh PLR adalah dengan rumus sebagai berikut :

$$PLR = \frac{Pd}{Ps} \dots\dots\dots(4.2)$$

Dimana :  $0 \leq t \leq T$

Dengan :

Pd = Paket yang mengalami *drop* (paket)

Ps = Paket yang dikirim (paket)

T = Waktu simulasi (detik)

t = Waktu pengambilan sampel (detik)

Dengan menggunakan rumus di atas maka akan diperoleh PLR dari masing-masing percobaan. Tabel 4.4 adalah hasil dari perhitungan PLR pada 5 buah *node* dengan menggunakan protokol TCP dan SCTP :

#### 4.4.2.1 Perhitungan *PLR* 5 Buah *Node* TCP Dan SCTP

**Tabel 4.4** Hasil Penghitungan PLR Protokol TCP Dan SCTP Pada 5 Buah *Node*

Percobaan Ke-	TCP(PLR)	SCTP(PLR)
1	0.057101025	0.019249382
2	0.057652912	0.019284814
3	0.058433647	0.019214224
4	0.057423038	0.021216005
5	0.056962952	0.019284537
6	0.057606946	0.01838512
7	0.057423038	0.018417821
8	0.053683386	0.019284537
9	0.052755442	0.019284815
10	0.050427602	0.019421162

Dari Tabel 4.4 dapat dilihat bahwa rata-rata PLR yang terjadi pada penerapan protokol TCP 5 buah *node* yang tersebar secara *random* adalah :

$$\text{Rata - rata PLR} = (\sum \text{PLR}/10) = (0.7597164/10) = 0.07597164\%$$

Dengan rumus standar deviasi: (Walpole, Ronald E., Raymond H *Myers*. 1995).

$$S = \sqrt{\frac{\sum (X_i - (\text{rata-rata PLR}))^2}{(n-1)}} = 0.0026738242149034\%$$



Interval rata-rata *PLR* tcp dihitung dengan cara:

$$\begin{aligned}\text{Interval} &= 0.07597164 \pm Z_{0.05/2} \cdot (0.0026738242149034 / \sqrt{10}) \\ &= 0.07597164 \pm Z_{0.025} \cdot (0.0026738242149034 / 3.16227766) \\ &= 0.07597164 \pm 1.96 \cdot (0.000845537) \\ &= 0.07597164 \pm 0.00165725252 \\ &= (0.07597164 - 0.00165725252; 0.07597164 + 0.00165725252) \\ &= (0.07431438748; 0.07762889252)\end{aligned}$$

Jadi, nilai *PLR* yang dihasilkan pada penerapan TCP dengan 5 buah *node* adalah sekitar 0.07431438748 s/d 0.07762889252%.

Sedangkan untuk SCTP :

$$\text{Rata-rata } PLR = (\sum PLR / 10) = (0.294983333 / 10) = 0.029498333\%$$

$$S = \sqrt{\frac{\sum (x_i - (\text{rata-rata } PLR))^2}{(n-1)}} = 0.00076823605922052\%$$

Interval rata-rata *PLR* sctp dihitung dengan cara:

$$\begin{aligned}\text{Interval} &= 0.029498333 \pm Z_{0.05/2} \cdot (0.00076823605922052 / \sqrt{10}) \\ &= 0.029498333 \pm Z_{0.025} \cdot (0.00076823605922052 / 3.16227766) \\ &= 0.029498333 \pm 1.96 \cdot (0.000242937) \\ &= 0.029498333 \pm 0.00047615652 \\ &= (0.029498333 - 0.00047615652; 0.029498333 + 0.00047615652) \\ &= (0.02902217648; 0.02997448952)\end{aligned}$$

Jadi, nilai *PLR* yang dihasilkan pada penerapan SCTP dengan 5 buah *node* adalah sekitar 0.02902217648 s/d 0.02997448952%

#### 4.4.2.2 Perhitungan *PLR* 10 Buah *Node* TCP Dan Sctp

**Tabel 4.5** Hasil Penghitungan *PLR* Protokol TCP Dan Sctp Pada 10 Buah *Node*

Percobaan Ke-	TCP( <i>PLR</i> )	Sctp( <i>PLR</i> )
1	0.067162377	0.025610196
2	0.067117335	0.025895708
3	0.067072287	0.025970903
4	0.067117335	0.025375827
5	0.06729748	0.025727338
6	0.067432543	0.027912341
7	0.067387527	0.025493025
8	0.067342505	0.028024449
9	0.067117335	0.0252586
10	0.066215563	0.025141345

Dari Tabel 4.5 dapat dilihat bahwa rata-rata *PLR* yang terjadi pada penerapan protokol TCP 10 buah *node* yang tersebar secara *random* adalah :

$$\text{Rata - rata } PLR = (\sum PLR / 10) = (0.671262287 / 10) = 0.067126229\%$$

$$S = \sqrt{\frac{\sum (x_i - (\text{rata-rata } PLR))^2}{(n-1)}} = 0.00034512049668809\%$$

Interval rata-rata *PLR* tcp dihitung dengan cara:

$$\begin{aligned} \text{Interval} &= 0.067126229 \pm Z_{0.05/2} \cdot (0.0011782311545301 / \sqrt{10}) \\ &= 0.067126229 \pm Z_{0.025} \cdot (0.0011782311545301 / 3.16227766) \\ &= 0.067126229 \pm 1.96 \cdot (0.000372589) \end{aligned}$$

$$\begin{aligned}
&= 0.067126229 \pm 0.00073027444 \\
&= (0.067126229 - 0.00073027444 ; 0.067126229 + 0.00073027444) \\
&= (0.06639595456 ; 0.06785650344)
\end{aligned}$$

Jadi, nilai *PLR* yang dihasilkan pada penerapan TCP dengan 10 buah *node* adalah sekitar 0.06639595456 s/d 0.06785650344%.

Sedangkan untuk SCTP :

$$Rata - rata\ PLR = (\sum PLR/10) = (0.260409733/10) = 0.026040973\%$$

$$S = \sqrt{\frac{\sum (xi - (rata-rata\ PLR))^2}{(n-1)}} = 0.0010493514906101\%$$

Interval rata-rata *PLR* sctp dihitung dengan cara:

$$\begin{aligned}
Interval &= 0.026040973 \pm Z_{0.05/2} \cdot (0.0010493514906101/ \sqrt{10}) \\
&= 0.026040973 \pm Z_{0.025} \cdot (0.0010493514906101/ 3.16227766) \\
&= 0.026040973 \pm 1.96 \cdot (0.000331834) \\
&= 0.026040973 \pm 0.00065039464 \\
&= (0.026040973 - 0.00065039464; 0.026040973 + 0.00065039464) \\
&= (0.02539057836; 0.02669136764)
\end{aligned}$$

Jadi, nilai *PLR* yang dihasilkan pada penerapan SCTP dengan 10 buah *node* adalah sekitar 0.02539057836 s/d 0.02669136764%.

#### 4.4.2.3 Perhitungan *PLR* 20 Buah *Node* TCP Dan Sctp

**Tabel 4.6** Hasil Penghitungan *PLR* Protokol TCP Dan Sctp Pada 20 Buah *Node*

Percobaan Ke-	TCP( <i>PLR</i> )	Sctp( <i>PLR</i> )
1	0.075912963	0.031425622
2	0.075913239	0.031481163
3	0.075918034	0.031536697
4	0.075923104	0.031508931
5	0.070938172	0.031453393
6	0.072088431	0.029617075
7	0.075938034	0.02964495
8	0.071913304	0.031481163
9	0.075913099	0.023473636
10	0.075918021	0.023360703

Dari Tabel 4.6 dapat dilihat bahwa rata-rata *PLR* yang terjadi pada penerapan protokol TCP 20 buah *node* yang tersebar secara *random* adalah :

$$\text{Rata - rata } PLR = (\sum PLR / 10) = (0.746376401 / 10) = 0.07463764\%$$

$$S = \sqrt{\frac{\sum (x_i - (\text{rata-rata } PLR))^2}{(n-1)}} = 0.0020845772399629\%$$

Interval rata-rata *PLR* tcp dihitung dengan cara:

$$\text{Interval} = 0.07463764 \pm Z_{0.05/2} \cdot (0.0020845772399629 / \sqrt{10})$$

$$= 0.07463764 \pm Z_{0.025} \cdot (0.0020845772399629 / 3.162277660168379)$$

$$= 0.07463764 \pm 1.96 \cdot (0.000659201)$$

$$\begin{aligned}
&= 0.07463764 \pm 0.00129203396 \\
&= (0.07463764 - 0.00129203396 ; 0.07463764 + 0.00129203396) \\
&= (0.07334560604 ; 0.07592967396)
\end{aligned}$$

Jadi, nilai *PLR* yang dihasilkan pada penerapan TCP dengan 20 buah *node* adalah sekitar 0.07334560604 s/d 0.07592967396%.

Sedangkan untuk SCTP :

$$Rata - rata PLR = (\sum PLR/10) = (0.294983333/10) = 0.029498333\%$$

$$S = \sqrt{\frac{\sum (xi - (rata-rata PLR))^2}{(n-1)}} = 0.0032930977714504\%$$

Interval rata-rata *PLR* sctp dihitung dengan cara:

$$\begin{aligned}
Interval &= 0.029498333 \pm Z_{0.05/2} \cdot (0.0032930977714504/ \sqrt{10}) \\
&= 0.029498333 \pm Z_{0.025} \cdot (0.0032930977714504/ 3.16227766) \\
&= 0.029498333 \pm 1.96 \cdot (0.00104136) \\
&= 0.029498333 \pm 0.0020410656 \\
&= (0.029498333 - 0.0020410656; 0.029498333 + 0.0020410656) \\
&= (0.0274572674; 0.0315393986)
\end{aligned}$$

Jadi, nilai *delay* yang dihasilkan pada penerapan SCTP dengan 20 buah *node* adalah sekitar 0.0274572674 s/d 0.0315393986%.

#### 4.5 Analisis Perbandingan Unjuk Kerja Protokol TCP Dan SCTP

Dari hasil perhitungan parameter-parameter QOS telah menghasilkan nilai parameter yang dapat menjadikan acuan untuk penerapan protokol TCP dan SCTP.

Dari hasil-hasil tersebut dapat diketahui data-data sebagai berikut :

**Tabel 4.7** Jumlah *Delay* Dan Rata-rata *Delay* Protokol TCP Dan SCTP

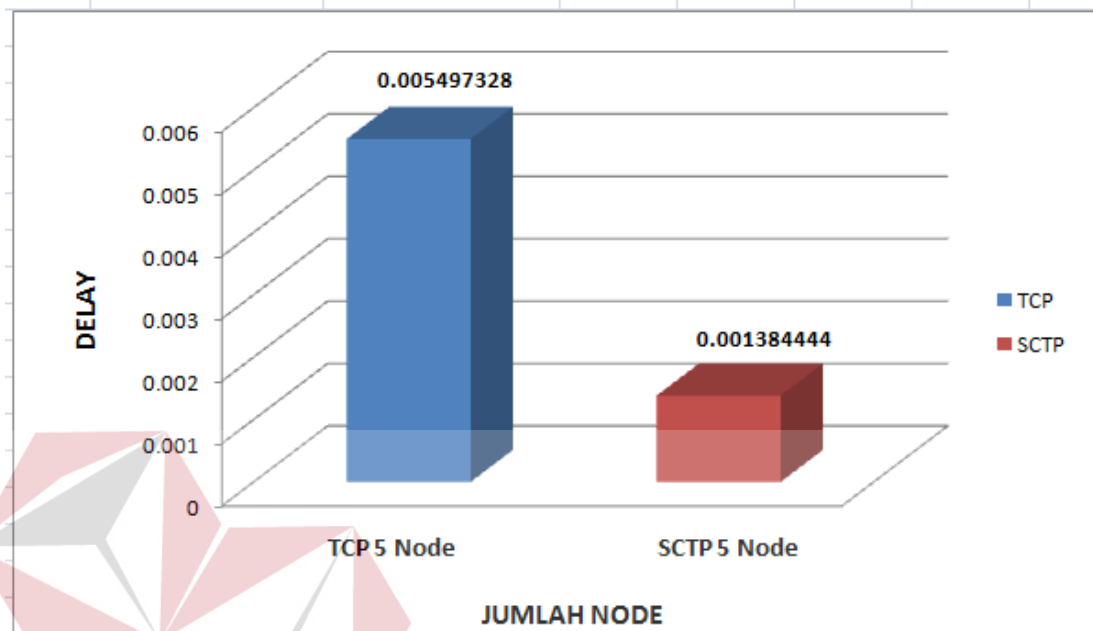
Jumlah <i>Node</i>	Protokol		Protokol	
	Jumlah Delay	Jumlah Delay	Rata-rata Delay	Rata-rata Delay
	TCP	SCTP	TCP	SCTP
5	0.054973278	0.013844435	0.005497328	0.001384444
10	0.069273278	0.042448435	0.006927328	0.004244844
20	0.084817967	0.081514731	0.008481797	0.008151473

**Tabel 4.8** Jumlah PLR Dan Rata-rata PLR Protokol TCP Dan SCTP

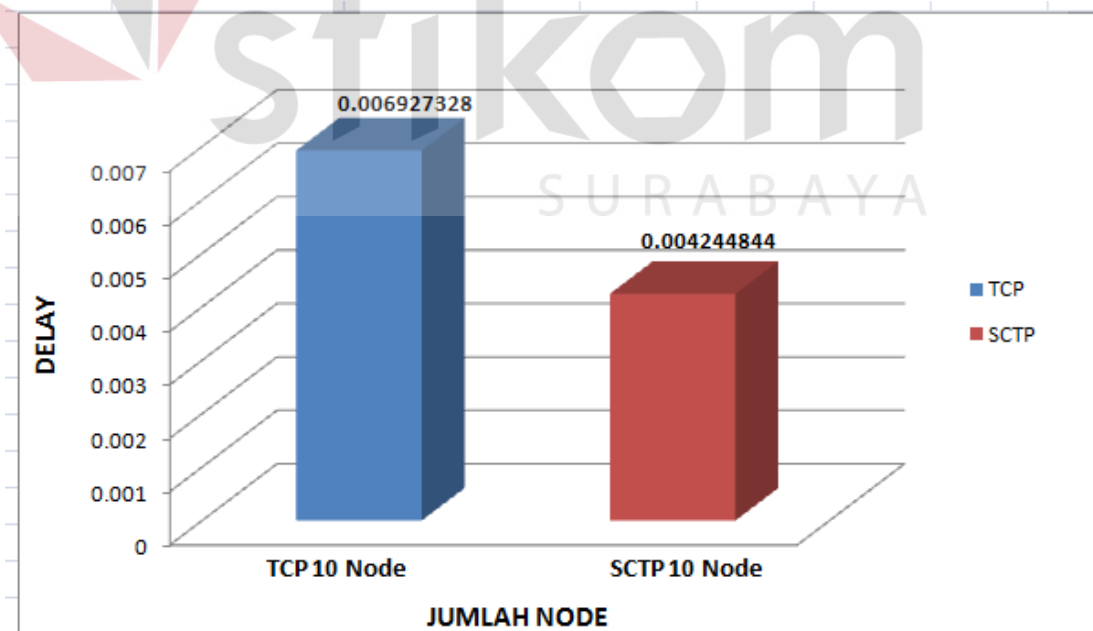
Jumlah <i>Node</i>	Protokol		Protokol	
	Jumlah PLR TCP	Jumlah PLR	Rata-rata PLR	Rata-rata PLR
		SCTP	TCP	SCTP
5	0.559469987	0.193042416	0.055946999	0.019304242
10	0.671262287	0.260409733	0.067126229	0.026040973
20	0.746376401	0.294983333	0.07463764	0.029498333

Tabel 4.7 dan Tabel 4.8 merupakan isi dari jumlah dan rata-rata seluruh parameter-parameter QOS yang dilakukan dalam percobaan. Dari data-data tersebut di dapat berupa gambaran dalam bentuk grafik sebagai berikut :

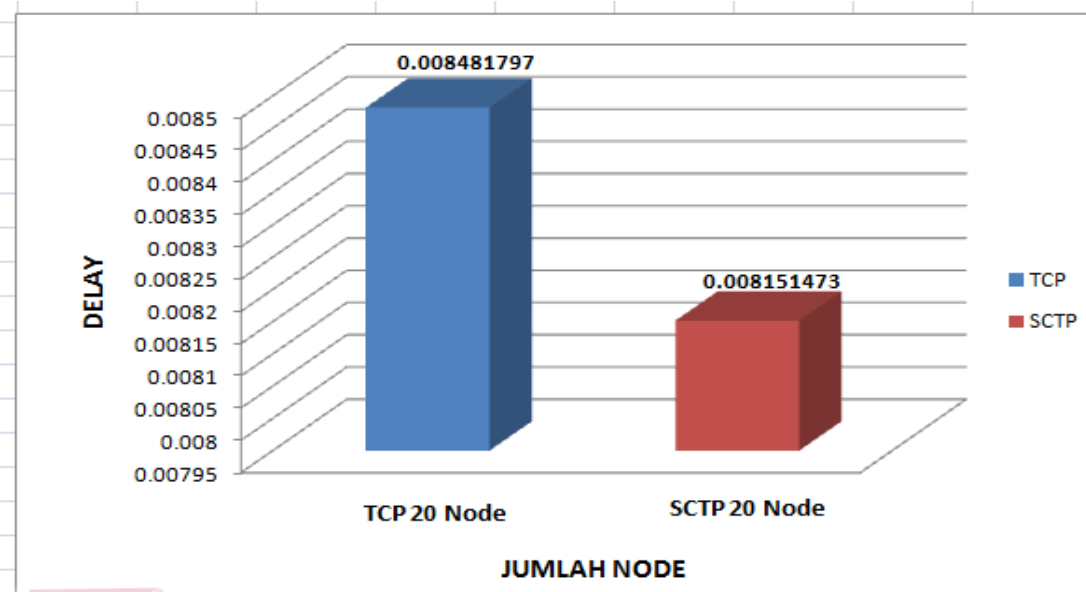
#### 4.5.1 Hasil Grafik Rata-rata *Delay* Protokol TCP & SCTP 5, 10, & 20 Buah *Node*



Gambar 4.15 Grafik Hasil Rata-rata *Delay* Protokol TCP & SCTP 5 Buah *Node*

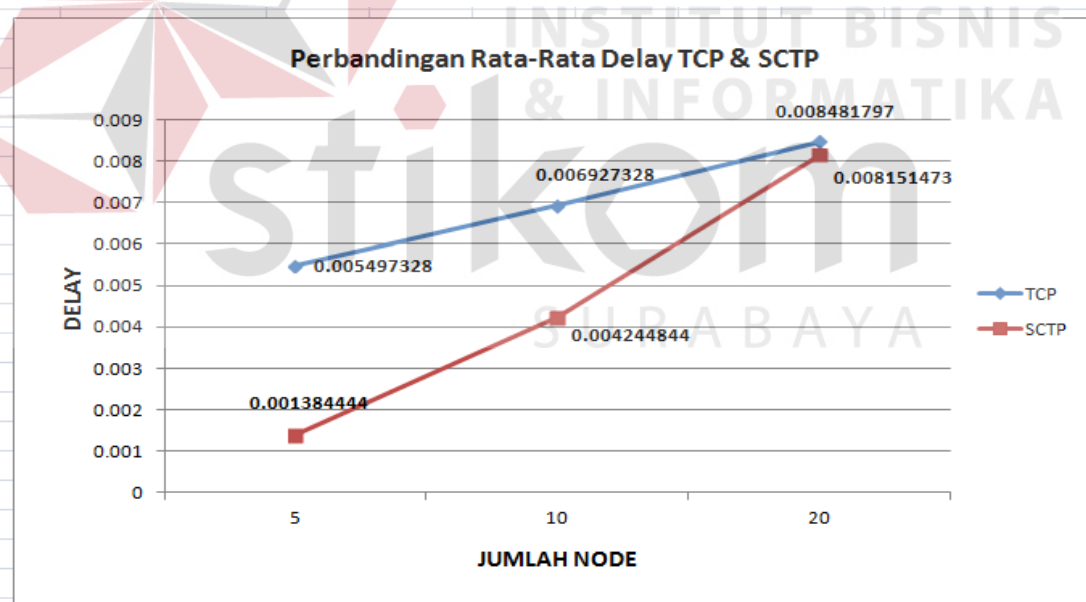


Gambar 4.16 Grafik Hasil Rata-rata *Delay* Protokol TCP & SCTP 10 Buah *Node*



Gambar 4.17 Grafik Hasil Rata-rata Delay Protokol TCP & Sctp 20 Buah Node

#### 4.5.2 Analisis Perbandingan Delay



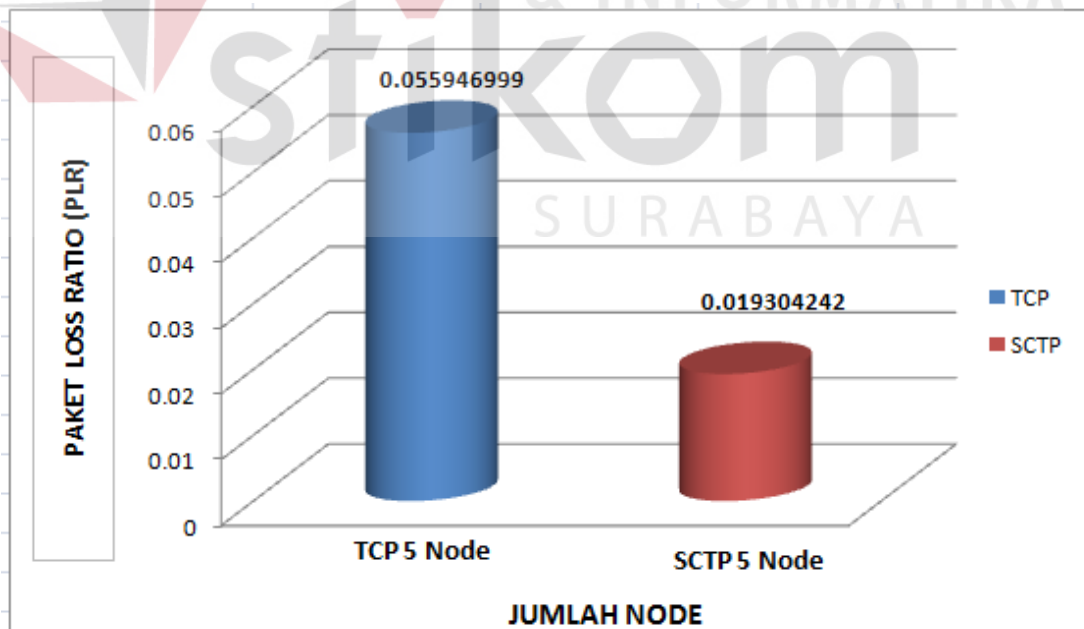
Gambar 4.18 Grafik Rata-rata Delay Protokol TCP Dan Sctp 5, 10 dan 20 Buah Node

Dari grafik pada Gambar 4.18 dapat diketahui bahwa rata-rata *delay* yang terjadi pada saat penerapan protokol TCP mulai dari 5 buah *node*, 10 buah *node*, dan 20 buah *node* lebih besar dibandingkan dengan penerapan protokol Sctp. Dimana

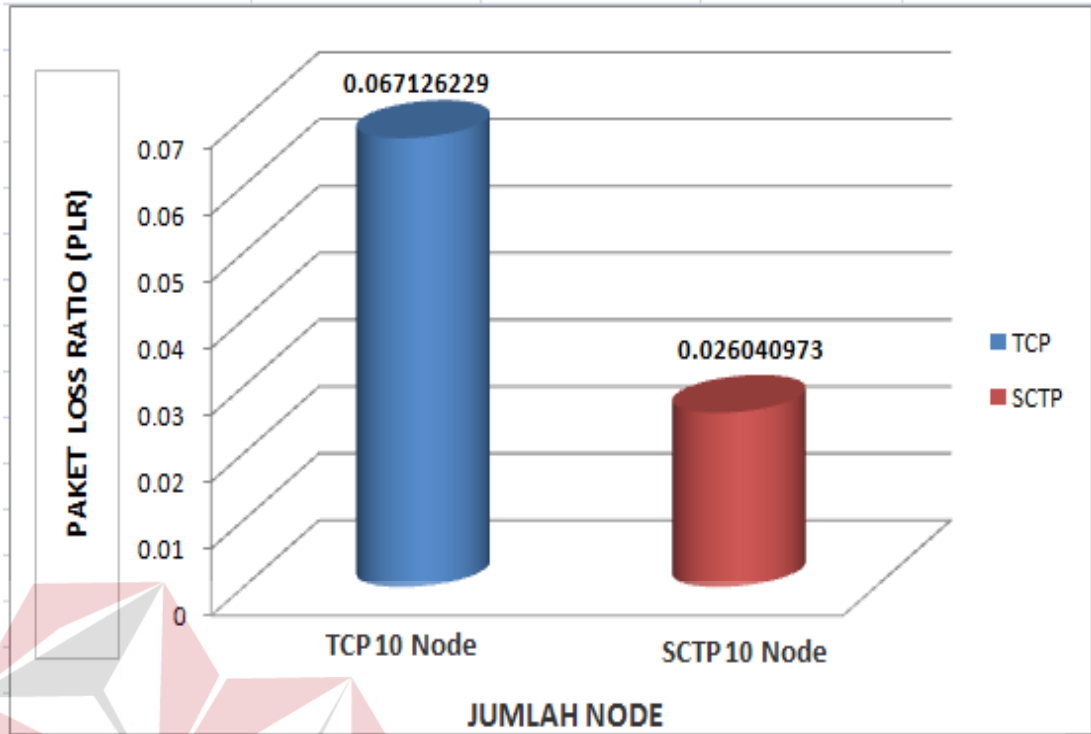


rata-rata *delay* dari penerapan protokol TCP 5 buah *node* adalah 0.0543226560 detik, TCP 10 buah *node* adalah 0.006927328 detik dan 0.008481797 detik untuk TCP 20 buah *node*. Sedangkan rata-rata *delay* dari penerapan protokol SCTP adalah 0.01308089 detik untuk 5 buah *node*, 0.004244844 detik untuk 10 buah *node*, dan 0.008151473 detik untuk 20 buah *node*. Hal ini dikarenakan proses pengiriman protokol SCTP menggunakan *multistream*. Dimana *multistream* pada protokol SCTP adalah mengirimkan data lebih dari satu *stream* sedangkan pada protokol TCP tidak memiliki mekanisme *multistream* sehingga jika terjadi masalah dalam proses pengiriman informasi pada jaringan maka SCTP lebih unggul di bandingkan TCP dikarenakan SCTP langsung menggunakan mekanisme *multistream* dalam mencari *trouble* jaringan.

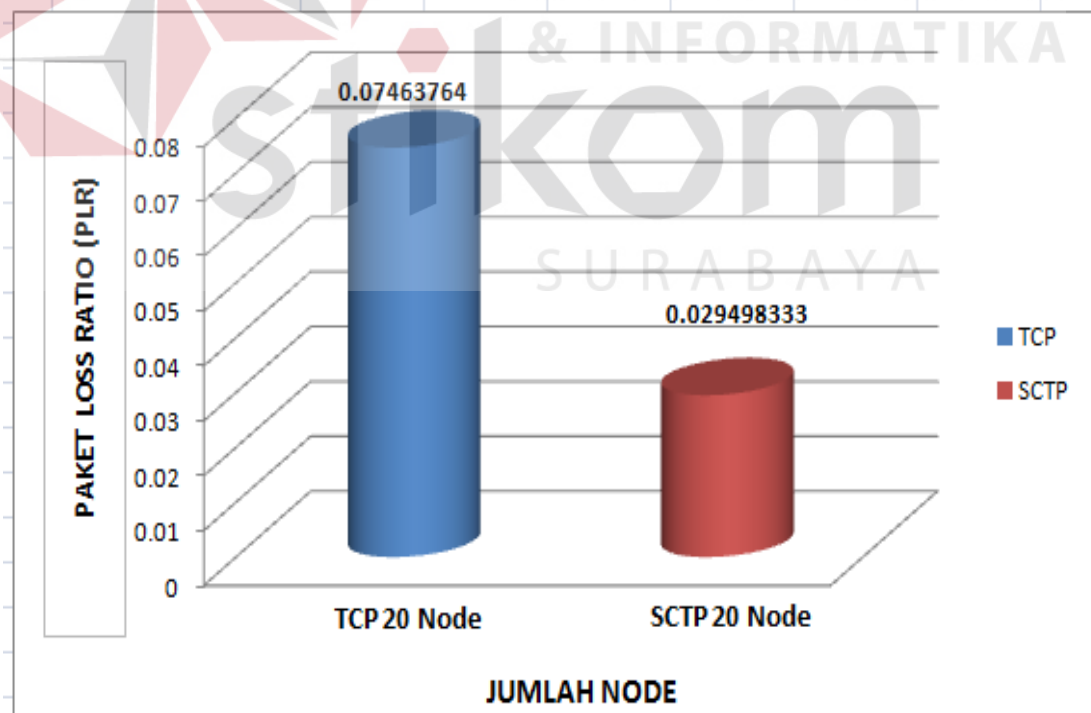
#### 4.5.3 Hasil Grafik Rata-rata *PLR* Protokol TCP & SCTP 5, 10, & 20 Buah



**Gambar 4.19** Grafik Rata-rata *PLR* Protokol TCP Dan SCTP 5 Buah *Node*

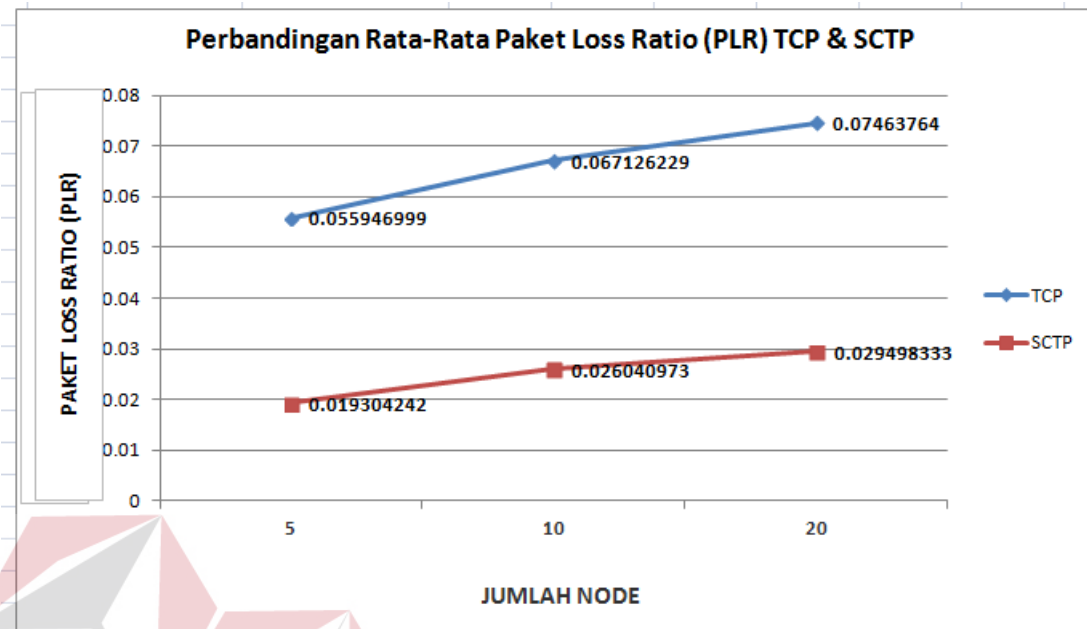


Gambar 4.20 Grafik Rata-rata *PLR* Protokol TCP Dan Sctp 10 Buah *Node*



Gambar 4.21 Grafik Rata-rata *PLR* Protokol TCP Dan Sctp 20 Buah *Node*

#### 4.5.4 Analisis Perbandingan *PLR*



**Gambar 4.22** Grafik Rata-rata *PLR* Protokol TCP Dan SCTP 5, 10 dan 20 Buah *Node*

Dari grafik pada Gambar 4.22 dapat diketahui bahwa rata-rata *PLR* yang terjadi pada saat penerapan protokol TCP mulai dari 5 buah *node*, 10 buah *node*, dan 20 buah *node* lebih besar dibandingkan dengan penerapan protokol SCTP. Dimana rata-rata *PLR* dari penerapan protokol TCP 5 buah *node* adalah 0.055946999%, TCP 10 buah *node* adalah 0.067126229% dan 0.07463764% untuk TCP 20 buah *node*. Sedangkan rata-rata *PLR* dari penerapan protokol SCTP adalah 0.019304242% untuk 5 buah *node*, 0.026040973% untuk 10 buah *node*, dan 0.029498333% untuk 20 buah *node*. Hal ini dikarenakan proses pengiriman protokol SCTP menggunakan *multihoming*. Dimana *multihoming* pada protokol SCTP adalah proses *redundance* jika terjadi kegagalan pada jaringan sedangkan pada protokol TCP tidak memiliki mekanisme *multihoming* sehingga jika terjadi kegagalan dalam proses pengiriman informasi pada jaringan secara bersamaan maka SCTP lebih unggul di banding TCP

dikarenakan SCTP langsung menggunakan mekanisme *multihomingnya* dalam mencari *alternative* lain tanpa harus menunggu sampai jaringan tersebut *up* kembali.

#### **4.6 Analisis Hasil Pengukuran Terhadap Penambahan Jumlah Node**

##### **4.6.1 Analisis Hasil Pengukuran Terhadap Penambahan Jumlah Node Dilihat Dari Sisi Parameter Delay**

Dari hasil pengukuran berdasarkan parameter *delay* yang digunakan dapat diketahui bahwa *node* dengan jumlah 5 buah pada protokol TCP dan SCTP menghasilkan nilai rata-rata *delay* sebesar 0.005497328 detik untuk protokol TCP sedangkan untuk SCTP adalah sebesar 0.001384444 detik. Pada penambahan jumlah *node* menjadi 10 buah *node* pada protokol TCP dan SCTP nilai rata-rata *delay* adalah sebesar 0.006927328 detik untuk TCP dan 0.004244844 detik untuk SCTP. Selanjutnya pada penambahan jumlah *node* sebesar 20 buah baik pada protokol TCP dan SCTP dengan nilai rata-rata *delay* 0.008481797 detik TCP dan 0.008151473 detik untuk SCTP. Dengan demikian dapat diketahui pada protokol TCP dan SCTP pada 5 buah *node* terjadi perbedaan jarak *delay* sebesar 0.004112884 detik. Pada 10 buah *node* terjadi perbedaan jarak *delay* sebesar 0.002682484 detik. Sedangkan pada penambahan jumlah *node* 20 jarak *delay* yang terjadi hanya sekitar 0.000330324 detik. Hal ini disebabkan karena pada penambahan jumlah *node* yang ke 20 pada protokol SCTP terjadi kegagalan di dalam jaringan pada saat percobaan simulasi dijalankan, sehingga pada protokol SCTP terjadilah proses *multistream* yang dimiliki oleh protokol SCTP. Dimana jika terjadi kegagalan didalam jaringan SCTP dengan kemampuan *multistreamnya* yaitu mengirimkan data lebih dari satu *stream*, sehingga SCTP memilih jalur *alternative* yang lain untuk proses pengiriman data ke tujuannya

agar biasa sampai dengan baik. Meskipun demikian dapat dilihat juga perbedaan jarak *delay* pada protokol TCP dan SCTP pada 20 buah *node* terlihat cukup dekat yaitu hanya sekitar 0.000330324 detik, itu dikarenakan *bandwidth* yang digunakan oleh sistem hanya sebesar 1.2 Mbps (*default* sistem) masih kurang untuk protokol SCTP. Karena pada *mobile node* data multimedia untuk protokol SCTP membutuhkan *bandwidth* yang cukup besar, karena pada data multimedia membutuhkan proses *real time* dimana data yang dikirimkan harus secepat dan seakurat mungkin agar bisa sampai dengan baik ke tujuannya.

#### **4.6.2 Analisis Hasil Pengukuran Terhadap Penambahan Jumlah *Node* Dilihat Dari Sisi Parameter *Packet Loss Ratio (PLR)***

Dari hasil pengukuran berdasarkan parameter *PLR* yang digunakan dapat diketahui bahwa *node* dengan jumlah 5 buah pada protokol TCP dan SCTP menghasilkan nilai rata-rata *PLR* sebesar 0.055946999% untuk protokol TCP sedangkan untuk SCTP adalah sebesar 0.019304242%. Pada penambahan jumlah *node* menjadi 10 buah *node* pada protokol TCP dan SCTP nilai rata-rata *PLR* menjadi sebesar 0.067126229% untuk TCP dan 0.026040973% untuk SCTP. Selanjutnya pada penambahan jumlah *node* sebesar 20 buah baik pada protokol TCP dan SCTP dengan nilai rata-rata *PLR* sebesar 0.07463764 % untuk TCP dan 0.029498333% untuk SCTP. Dengan demikian dapat diketahui pada protokol TCP dan SCTP pada 5 buah *node* terjadi perbedaan jarak *PLR* sebesar 0.036642757%. Pada 10 buah *node* terjadi perbedaan jarak *PLR* sebesar 0.041085256%. Selanjutnya pada penambahan jumlah *node* 20 jarak *PLR* yang terjadi sekitar 0.045139307%. Dengan demikian pada protokol TCP dan SCTP dilihat dari sisi *PLR* nilai yang dihasilkan sudah sesuai

dengan hasil pada percobaan yang telah dilakukan, sehingga pada sisi *PLR* ini tidak terjadi kegagalan jaringan seperti yang dialami oleh parameter *delay*. Maka dari itu pada parameter *PLR* kemampuan protokol SCTP yaitu (*multistream*), dimana data dikirimkan lebih dari satu *stream* jika terjadi kegagalan dalam jaringan dapat berjalan dengan baik karena tidak terjadi kegagalan dalam percobaan pada sisi *PLR* baik protokol TCP dan Sctp.

