

# Implementasi *Load Balancing Peer Connection Classifier (Pcc)* Pada Jaringan Intenet di Rumah Sakit Umum Daerah Prabumulih

Alan Fauzi<sup>1</sup>, Alex Wijaya<sup>2</sup>, Irman Effendy<sup>3</sup>

<sup>1)</sup> Mahasiswa Informatika Universitas Bina Darma

<sup>2)</sup> Dosen Ilmu Komputer.

Jl Jend A. Yani No.12 Plaju,

Palembang 30264

Email: alanfauzi70@yahoo.com<sup>1)</sup> alexwijaya@mail.binadarma.ac.id<sup>2)</sup>,

irman.efenddy@mail.binadarma.ac.id<sup>3)</sup>

**Abstrak.** *Load Balancing* merupakan teknik untuk mendistribusikan beban *traffic* pada dua atau lebih jalur koneksi secara seimbang, agar *traffic* dapat berjalan optimal, memaksimalkan *throughput*, memperkecil waktu tanggap dan menghindari *overload* pada satu jalur koneksi. dimana kepadatan jalur menjadi perhitungan pertama dalam pembagian *bandwidth*. Router dapat mendistribusikan lalu lintas di beberapa jalur ke tujuan yang sama, menyeimbangkan beban di sumber jaringan yang berbeda. Sebuah penyeimbangan beban *server*, pada sebaliknya mendistribusikan lalu lintas antar sumber daya *server* dari pada jaringan sumber daya. *PCC* adalah teknik dalam pengelompokan *traffic* koneksi yang melalui atau keluar masuk router menjadi beberapa kelompok yang dibedakan berdasarkan *src-address*, *dst-address*, *src-port* dan *dst-port*. Agar pada router akan mengingat jalur *gateway* yang dilewati awal *traffic* koneksi. jadi hasil dari pengembangan *load balancing pcc* dapat meningkatkan kecepatan koneksi dan membagi beban *traffic* pada kedua *gateway* agar tidak terjadi *overload*. Penulis juga menerapkan teknik *fail over*, jika salah satu koneksi *gateway* sedang terputus maka *gateway* yang lainnya otomatis akan menampung semua *traffic*.

## 1. Pendahuluan

### 1.1 Latar Belakang

*Load balancing* merupakan teknik untuk mendistribusikan beban *traffic* pada jalur atau lebih dari dua jalur koneksi secara seimbang, agar *traffic* dapat berjalan secara optimal, memaksimalkan *throughput*, memperkecil waktu tanggap dan menghindari kegagalan pada salah satu jalur koneksi. Solusi jaringan ini untuk mendistribusikan lalu lintas masuk antara *server* hosting konten aplikasi yang sama.

Dengan menyeimbangkan permintaan aplikasi di beberapa *server*, menyeimbangkan beban untuk mencegah *server* aplikasi dari titik kegagalan

*Peer Connection Classifier (PCC)* adalah suatu teknik dalam pengelompokan *traffic* koneksi yang keluar masuk melalui *router* menjadi beberapa kelompok. Untuk dibedakan berdasarkan *src-address*, *dst-address*, *src-port* dan *dst-port*. *Router* dapat mengingat jalur *gateway* yang dilewati dari awal *traffic* koneksi, sehingga paket-paket selanjutnya yang masih berkaitan dengan koneksi awal akan dilewatkan pada jalur *gateway* yang sama. Kelebihan dari PCC dapat menjawab banyaknya keluhan sering terjadi putusnya koneksi pada teknik *load balancing* lainnya sebelum adanya PCC karena perpindahan *gateway*. Dalam mengatasi kelebihan beban pada *server* hal yang paling utama dengan meningkatkan kualitas atau kecanggihan sebuah *server* misalnya meng-*upgrade* *cpu* dan menambah memori. Menggunakan lebih dari satu *server* dengan harga dan kualitas rata-rata seringkali jauh lebih dari efektif dan menguntungkan dari pada menggunakan sebuah *server* mahal yang berkinerja tinggi. Namun solusi banyak *server* ternyata bukan tanpa masalah. Maka dari itu akan diterapkan sistem *load balancing*.

Rumah Sakit Umum Daerah Prabumulih merupakan salah satu institusi yang bergerak dalam bidang pelayanan kesehatan kepada masyarakat. Karyawan serta staff rumah sakit lainnya telah memanfaatkan jaringan internet sebagai media pendukung dalam kegiatan institusi mereka, seperti pemanfaatan sarana dan prasarana yang tersedia. Namun dengan kebutuhan akan proses informasi yang cepat, maka berdampaklah pada banyaknya beban (*traffic*) serta permintaan (*request*), sehingga *broadcast* dalam suatu jaringan semakin meningkat dan akan berdampak pada optimalnya kinerja dari jaringan itu sendiri, Adapun batasan permasalahan dalam penelitian ini sebagai berikut :

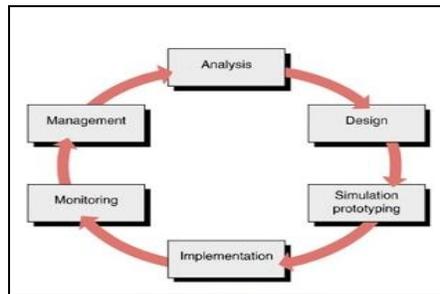
1. Bagaimana Mengimplementasi *load balancing peer connection classifier (pcc)* pada jaringan internet di rumah sakit umum daerah prabumulih ?.

## **2.1 Metode Penelitian**

Metode yang digunakan dalam penelitian ini adalah Deskriptif

## **3.1 Metode Pengembangan Sistem**

Metode Pengembangan Sistem yang digunakan dalam penelitian ini adalah *Network Development Life Cycle (NDLC)*.



Gambar 1 Siklus NDLC

Berikut penjelasan dari masing-masing tahapan pada NDLC:

- a. **Analisis** : melakukan analisa terhadap system yang telah berjalan untuk menentukan metode *Load Balancing* dan mengumpulkan data sebuah jaringan dengan menggunakan mikrotik sebagai *Load Balancing*.
- b. **Design** : perancangan untuk melakukan *desain* adalah Topologi *Star* yang melakukan pembagian beban *bandwidth* yang merata serta proses penentuan alur *traffic* antar dua ISP yang ada.
- c. **Simulation prototyping** : simulasi dengan menggunakan *virtual machine* yang berdasarkan rancangan penelitian terdapat *user*, mikrotik sebagai *Load Balancing* dan dua buah ISP berbeda.
- d. **Implementation** : pada tahap ini akan diterapkan semua yang telah direncanakan dan dirancang untuk melakukan pengujian apakah berhasil dengan topologi jaringan yang telah didesain sebelumnya.
- e. **Monitoring** : perlunya pemantauan terhadap *traffic* di jaringan supaya bisa berjalan sesuai dengan analisa awal dan pemantauan terhadap infrastruktur *hardware*.
- f. **Management** : tahapan ini memerlukan perhatian khusus terhadap kebijakan yang perlu dibuat untuk mengatur dan membuat sistem dapat berjalan dengan baik.

## 4. Hasil dan Pembahasan

### 4.1.1 Konfigurasi Mikrotik

Setelah *interface* yang telah terpasang telah terbaca oleh mikrotik OS maka dapat memulai melakukan konfigurasi mikrotik mulai dari konfigurasi *IP address*, *gateway*.

```

/ip address
add address=192.168.1.1/24 interface=ether1
comment=ISP 1 for IIX disabled=no
add address=192.168.2.2/24 interface=ether2
comment=ISP 2 for internasional disabled=no
add address=171.16.21.1/24 interface=ether3
disabled=no
  
```

Dijelaskan memasukan *ip address* kita pada *interface* yang telah kita tentukan

```
/ip gateway
ip route add gateway=192.168.2.2
ip route add gateway=192.168.1.1
```

Dijelaskan kita gunakan untuk memasukan *gateway* yang telah ditentukan

#### 4.1.2 Konfigurasi *Load Balancing Pcc*

Setelah melakukan konfigurasi mikrotik selanjutnya kita dapat melakukan konfigurasi *load balancing pcc*.

##### a. Konfigurasi *Mangle*

```
/ip firewall mangle
add action=mark-connection chain=prerouting \
comment="" connection-state=new disable=no \
dst-port=80 in-interface=LOKAL \
new-connection-mark=1b_1 pcc=2,1 protokol=tcp \
passthrough=yes \
add action=mark-routing chain=prerouting comment="" \
comment="" connection-state=new disable=no \
in-interface=LOKAL new-routing-mark=route_1 \
passthrough=no
```

##### b. Konfigurasi *Routing*

```
/ip route
add check-gateway=ping comment="default gateway"
disabled=no distance=1 dst-address=0.0.0.0/0 \
gateway=192.168.2.2 scope=255 target-scope=10
```

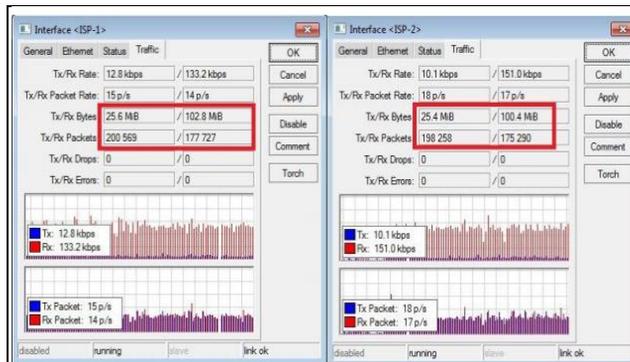
##### c. Konfigurasi NAT

```
/ip firewall nat
add action=masquerade chain=srcnat comment="" \
disabled=no out-interface=ISP-1
add action=masquerade chain=srcnat comment="" \
disabled=no out-interface=ISP-2
```

## 4.2 Hasil

### 4.2.1 Pengujian *Load Balancing Pcc*

Setelah melakukan konfigurasi pada mikrotik selanjutnya dapat memulai melakukan pengujian *load balancing pcc*. Pada tahapan ini penulis memonitoring system jaringan dengan menggunakan aplikasi atau *tools* yang ada pada *winbox*. Hasil monitoring pada menu *interface list*.



Gambar 2 Grafik koneksi pada setiap gateway

Dari gambar diatas dapat dilihat dari kedua *traffic* di *interface* adalah rata-rata penyebaran dari tiap-tiap *gateway* ISP. Pada *interface* ISP-1 dan *interface* ISP-2 terlihat dari besar *packet* dan *bytes* yang telah dilewati.

#### 4.2.2 Membuat Pengaturan “Fail Over”

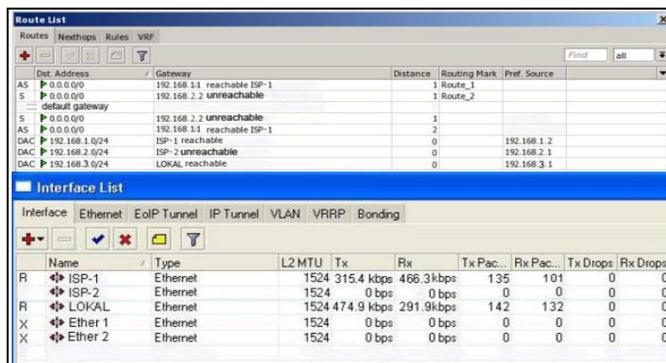
Dalam *load balancing fail over* adalah salah satu *gateway* sedang di diskoneksikan, maka *gateway* lainnya otomatis akan menjadi *default gateway* yang menampung semua *traffic* jaringan.

```

/ip route
add check-gateway=ping comment="" disabled=no \
distance=2 dst-address=0.0.0.0/0 \
gateway=192.168.1.2 scope=225 target-scope=10

```

Berikut ini hasil dari pengaturan “fail over” jika salah satu koneksi dalam keadaan mati.



Gambar 3 Teknik fail over

## **5. Kesimpulan**

### **5.1 Kesimpulan**

Dari hasil penelitian mengenai *load balancing peer connection classifier (pcc)* dapat disimpulkan sebagai berikut :

1. Penerapan *load balancing pcc* telah memberikan *bandwidth* yang optimal, namun *load balancing* tidak dapat mengakumulasi *bandwidth* kedua koneksi.
2. Penerapan *load balancing pcc* telah membagi beban *traffic* secara seimbang pada ISP-1 dan ISP-2 pada rumah sakit umum daerah prabumulih.
3. Penerapan teknik *fail over* dapat menjadikan salah satu *gateway* sebagai koneksi tunggal jika *gateway* yang lain keadaan mati.

### **Referensi**

1. Nurul Fadilah Zamazami, Implementasi Load Balancing dan Failover Menggunakan Mikrotik RouterOS Berdasarkan Multihomed Gateway pada Warung Internet "DIGA". Politeknik Telkom, Bandung(2009).
2. Mohd Siddik, Yopi Hendro, Zulfian Azmi, Load Balance dan Pembagian Bandwidth Jaringan LAN Menggunakan Mikrotik Router Board RB 750. STMIK Triguna Dharma, Medan(2015).