

ANALISIS PERBANDINGAN QUALITY OF SERVICE PADA ROUTEROS MIKROTIK DAN OPENWRT DENGAN MENGGUNAKAN METODE LOAD BALANCING

Dika Handika¹, Mochamad Adhari Adiguna²

¹²Teknik Informatika, Universitas Pamulang, Banten, Indonesia, 15310
e-mail: ¹dikah53@gmail.com, ²dosen01864@unpam.ac.id

Abstract

With the number of internet users around the world increasing significantly, the demand for bandwidth is also increasing. This has led to an increase in heavy internet traffic. To optimize the use of available bandwidth, it is necessary to implement a load balancing mechanism. One effective mechanism is to use load balancing, which functions to level the bandwidth usage load in a balanced manner. This research applies load balancing with the Per Connection Classifier method on MikroTik and OpenWrt routers. The research method used is by collecting data through observation, interviews, and literature studies, and conducting system development methods using the NDLC (Network Development Life Cycles) method. The results of the research all load balancing methods on MikroTik and OpenWrt function properly. Load balancing performance in testing with the Internet Speed Tester scenario, the PCC method on OpenWrt is better than the PCC method load balancing performance on MikroTik routerOS based on ping, upload and download calculations. However, the jitter value on the MikroTik routerOS is better than the jitter value on OpenWrt. Based on TIPHON standards in tests with a 100 Mb file download scenario, MikroTik routerOS and OpenWrt have the same results, namely having a throughput value in the "medium" category with index 2 and having packet loss, delay and jitter values in the "very good" category with index 4.

Keywords : Load Balancing, PCC, MikroTik, OpenWrt, QoS

Abstrak

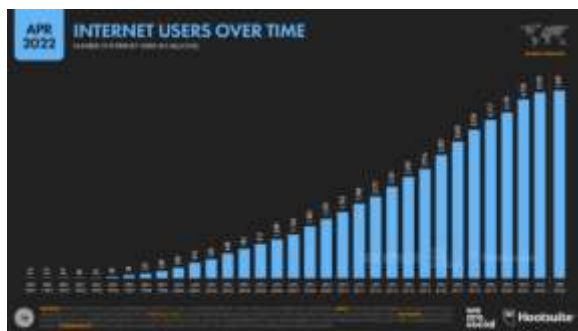
Dengan meningkatnya jumlah pengguna internet di seluruh dunia secara signifikan, permintaan akan *bandwidth* juga semakin meningkat. Hal ini menyebabkan peningkatan lalu lintas internet yang padat. Untuk mengoptimalkan penggunaan *bandwidth* yang tersedia, diperlukan penerapan mekanisme penyeimbangan beban. Salah satu mekanisme yang efektif adalah menggunakan *load balancing*, yang berfungsi untuk meratakan beban penggunaan bandwidth secara seimbang. Penelitian ini menerapkan *load balancing* dengan metode *Per Connection Classifier* pada routerOS MikroTik dan OpenWrt. Metode penelitian yang digunakan adalah dengan pengumpulan data melalui observasi, wawancara, dan studi pustaka, serta melakukan metode pengembangan sistem dengan menggunakan metode NDLC (*Network Development Life Cycles*). Hasil dari penelitian semua metode *load balancing* pada MikroTik maupun OpenWrt berfungsi dengan baik. Kinerja *load balancing* pada pengujian dengan skenario *Internet Speed Tester*, metode PCC pada OpenWrt lebih baik dari kinerja *load balancing* metode PCC pada routerOS MikroTik berdasarkan perhitungan *ping*, *upload* dan *download*. Akan tetapi, nilai *jitter* pada routerOS MikroTik lebih baik dibanding dengan nilai *jitter* pada OpenWrt. Berdasarkan standar TIPHON pada pengujian dengan skenario *download file* 100 Mb, routerOS MikroTik dan OpenWrt memiliki hasil yang sama, yaitu memiliki nilai *throughput* berkategori "sedang" dengan indeks 2 dan memiliki nilai *packet loss*, *delay* dan *jitter* berkategori "sangat bagus" dengan indeks 4.

Kata kunci: Load Balancing, PCC, MikroTik, OpenWrt, QoS

1. PENDAHULUAN

Pada saat ini, internet merupakan sesuatu yang tidak bisa dipisahkan dari kehidupan sehari-hari. Semua informasi yang diperlukan bisa diperoleh melalui internet [1]. Kebutuhan internet saat ini sangat meningkat pesat, karena dalam semakin banyak penggunaan internet untuk mencari informasi, berita maupun digunakan dalam media komunikasi [2].

Oleh karena itu diperlukan manajemen *bandwidth* agar kualitas internet menjadi lebih baik. *Bandwidth* didefinisikan sebagai lebar pita yang menentukan kecepatan akses jaringan komputer [1]. Kebutuhan *bandwidth* setiap hari semakin meningkat seiring dengan bertambahnya jumlah pengguna internet, pengguna internet di seluruh dunia telah bertumbuh secara signifikan. Menurut data *We Are Social* dan *Hootsuite* (2022), pengguna internet meningkat dari 3640 juta di tahun 2017 menjadi 5001 juta pada bulan april 2022.



Gambar 1. Data pengguna internet di dunia
[Sumber: wearesocial.com]

Meningkatnya jumlah pengguna internet tidak diimbangi dengan penambahan *bandwidth* internet. Hal tersebut dikarenakan faktor biaya yang menjadi pertimbangan. Jumlah pengguna internet yang semakin banyak jika tidak diimbangi dengan penambahan *bandwidth* dapat memunculkan permasalahan bagi para pengguna. Permasalahan tersebut terkait dengan kecepatan akses internet yang dirasa lama oleh para pengguna [3].

Filosofi Net adalah penyedia layanan internet dengan konsep RT/RW Net yang baru merintis sejak pandemi Covid-19. Berdasarkan keterangan pemiliknya, semakin lama klien-klien baru semakin bertambah. Hal ini menyebabkan beban *bandwidth* yang digunakan semakin besar. Maka dari itu, pemilik ingin menambah *bandwidth* yang digunakan sekaligus sebagai *backup* internet bila sumber utama mati. Dengan adanya permasalahan tersebut, maka perlu diadakan penyeimbangan beban, salah satu mekanisme

untuk mengoptimalkan *bandwidth* yang ada adalah dengan menggunakan *load balancing* yang akan menyeimbangkan beban *bandwidth* yang digunakan oleh penggunanya [4].

Load balancing adalah sebuah teknik mendistribusikan beban *traffic* pada dua jalur atau lebih, sehingga didapatkan sambungan yang seimbang, *traffic* yang lebih optimal, *throughput* data maksimal, *delay* minimal, serta tidak terjadi *overload* [5].

Penerapan *load balancing* memerlukan *routerOS* MikroTik atau OpenWrt. Pemilik Filosofi Net ingin menggunakan perangkat yang stabil dan optimal untuk memuaskan para kliennya. Maka dari itu perlu melakukan penelitian perangkat mana yang paling baik digunakan pada Filosofi Net.. Sudah banyak penelitian mengenai *load balancing*, seperti penerapan *load balancing* metode *Per Connection Classifier* berbasis *router* MikroTik di PT. Asuransi Jiwa Nasional [6], implementasi *load balancing* menggunakan metode PCC (*Per Connection Classifier*) berbasis MikroTik pada SMK Tunas Harapan Jakarta[7], implementasi *load balancing* menggunakan Raspberry Pi dengan *routerOS* OpenWrt [8].

Penelitian-penelitian tersebut membahas tentang penerapan dan implementasi *load balancing*. Penelitian lebih lanjut mengenai kinerja perlu dilakukan untuk mendapatkan hasil perbandingan kinerja pada suatu sistem yang diimplementasikan. Oleh karena itu penelitian ini bertujuan untuk melakukan perbandingan kinerja *routerOS* MikroTik dengan OpenWrt menggunakan teknik *load balancing* dan dilakukan analisis *Quality of Service* (QoS) dengan parameter *throughput*, *packet loss*, *delay* dan *jitter* yang kemudian akan ditentukan kategori layanan dengan standar *Telecommunications and Internet Protocol Harmonization Over Network* (TIPHON) yang merupakan standar penilaian parameter QoS yang dikeluarkan oleh badan standar ETSI (*European Telecommunications Standards Institute*), kemudian dianalisis bagaimana kriteria jaringan yang diuji dan diambil kesimpulan dari hasil parameter-parameter tersebut. Untuk pengujiannya menggunakan *load balancing* metode PCC (*Per Connection Classifier*) pada *routerOS* MikroTik dan OpenWrt.

2. METODE

Metode Pengumpulan Data

a. Observasi

Observasi adalah sebuah metode pengumpulan informasi yang dilakukan dengan

cara pengamatan atau peninjauan langsung terhadap objek penelitian. Observasi dilaksanakan di Filosofi Net pada tanggal 14 sampai dengan 21 Juni 2023.

b. Wawancara

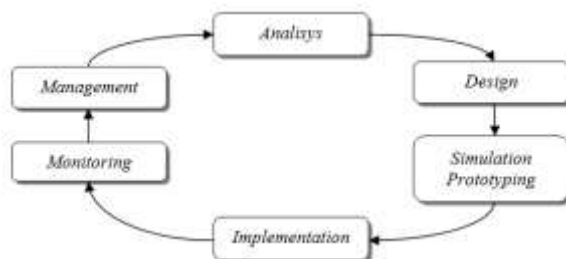
Metode ini dilakukan dengan cara mengadakan tanya jawab dengan pemilik dari Filosofi Net.

c. Studi Pustaka

Studi Pustaka merupakan penelusuran literatur yang berupa bacaan-bacaan kepustakaan seperti buku referensi, jurnal, paper dan bisa juga mengambil dari media online.

Metode Pengembangan Sistem

Metode pengembangan sistem yang akan digunakan adalah NDLC (Network Development Life Cycle). Metode ini dapat memberikan gambaran proses yang berupa tahapan-tahapan dari mekanisme teoritis yang dibutuhkan dalam suatu rangkaian proses pengembangan sistem pada jaringan komputer. Berikut merupakan tahapan metode NDLC:



Gambar 2. Tahapan NDLC
(Goldman dan Rawles, 2011)

Penjelasan dari tahapan NDLC pada Gambar 1 diatas adalah sebagai berikut:

a. Analysis (Analisis)

Model pengembangan sistem NDLC dimulai dengan fase analisis. Pada tahap ini dilakukan perumusan masalah hingga mengidentifikasi konsep, pemahaman tahap ini meliputi:

- 1) *Identify*, mengidentifikasi permasalahan yang dihadapi pada jaringan yang akan dibangun.
- 2) *Understand*, memahami mekanisme kerja sistem yang akan dibangun dan digunakan, dianalisis atau dikembangkan khususnya pada QoS jaringan dalam memberikan solusi dalam permasalahan yang dihadapi.
- 3) *Analyze*, menganalisis sejumlah komponen dan kebutuhan sistem yang dibangun dan dikembangkan.

4) *Report*, Kegiatan ini mempresentasikan proses hasil analisis dari jaringan yang akan dibangun.

b. Design (Perancangan)

Dalam tahap perancangan, langkah utamanya adalah menentukan cara sistem dapat menyelesaikan masalah yang telah diidentifikasi pada tahap analisis sebelumnya. Tahap perancangan mencakup beberapa kegiatan, termasuk perancangan topologi jaringan.

c. Simulation Prototype (Prototipe Simulasi)

Untuk memberikan gambaran umum tentang proses komunikasi dan interaksi antar elemen sistem yang akan dibangun, peneliti akan membuat sebuah prototipe dari sistem tersebut. Prototipe ini akan berfungsi sebagai simulasi dan implementasi awal, sehingga peneliti dapat memahami cara kerja sistem secara keseluruhan dan mengetahui keterkaitan antar elemen di dalamnya.

d. Implementation (Implementasi)

Pada tahap ini, akan melakukan implementasi sistem yang telah dirancang dengan menggunakan perangkat keras dan lunak yang tersedia. Seluruh perangkat akan ditempatkan sesuai dengan topologi yang telah dibuat sebelumnya.

e. Monitoring (Pengawasan)

Dalam tahap ini, akan melakukan monitoring terhadap jaringan yang telah dibangun untuk mengevaluasi kualitas jaringan dan kinerja QoS yang diterapkan dengan menggunakan teknik *load balancing*.

f. Management

Pengelolaan, pemeliharaan, dan perawatan sistem dianggap sebagai bagian penting dalam tahap NDLC karena dilakukan bersamaan dengan aktivitas pengelolaan sistem, yang bertujuan untuk memaksimalkan penggunaan sistem secara efisien dan efektif, serta memastikan kemudahan, fleksibilitas, pengelolaan, dan pengembangan sistem yang lebih baik.

3. HASIL

Analisis Sistem

Peneliti membagi aktifitas pada tahap analisis ini menjadi beberapa fase, yaitu:

a. Identify (Identifikasi Masalah)

Seiring dengan bertambahnya klien Filosofi Net, maka perlu penambahan *bandwidth* juga untuk menjaga kualitas jaringan pada Filosofi Net. Berdasarkan informasi dari

owner Filosofi Net bahwa saat ini Filosofi Net masih menggunakan 1 ISP sebagai sumber internetnya.

Berdasarkan permasalahan tersebut, Filosofi Net ingin penulis melakukan penelitian untuk menemukan solusi permasalahan tersebut. Yaitu dengan membangun atau menerapkan load balancing dengan metode *Per Connection Classifier* pada jaringan Filosofi Net, serta mencari *router* mana yang paling baik untuk digunakan untuk *load balancing* di Filosofi Net.

b. *Understand* (Memahami Rumusan Masalah)

Hasil identifikasi rumusan masalah membutuhkan pemahaman yang baik agar dapat menghasilkan solusi yang tepat. Oleh karena itu dengan adanya studi pustaka melalui perpustakaan dan internet untuk mengumpulkan data dan informasi dari berbagai sumber dalam bentuk buku, makalah, literature, dan berbagai situs web mengenai topik permasalahan yang ada disana, hasilnya dapat digunakan untuk memahami permasalahan yang terjadi untuk merumuskan solusi efektifitas dalam menyelesaikan berbagai perumusan masalah. Dengan pemahaman ini penulis berusaha membangun dan mengimplementasikan sistem yang diharapkan dapat mengatasi berbagai perumusan permasalahan yang ada. Oleh karena itu penulis perlu memahami konsep-konsep jaringan load balancing beserta metode *Per Connection Classifier* dan juga memahami parameter QoS berdasarkan standar TIPHON.

c. *Analyze* (Menganalisa Elemen Sistem)

Dari hasil analisis sistem yang didapat dari hasil identifikasi masalah dan pemahaman masalah yang telah dilakukan penulis untuk memberikan solusi didapat beberapa elemen atau komponen yang akan dibutuhkan dalam penerapan sistem yang akan dibangun, maka penulis akan membangun jaringan load balancing dengan metode *Per Connection Classifier* dan membandingkan RouterOS mana yang paling baik digunakan sesuai dengan parameter QoS berdasarkan standar TIPHON.

d. *Report* (Melaporkan Hasil Analisis)

Proses akhir dari fase analisis adalah pelaporan rincian dari berbagai komponen atau elemen sistem yang dibutuhkan. Berbagai elemen atau komponen tersebut mencakup:

1) Spesifikasi yang akan dibangun

- Load Balancing
- QoS

2) Spesifikasi *software* yang akan digunakan

- GNS3

- Google Chrome
- Wireshark
- Microsoft Excel
- Winbox

3) Spesifikasi *hardware* yang digunakan

- MikroTik rb941-2nd
- Mi Router 4a Gigabit
- MikroTik rb750gr3
- Switch
- PC Admin

4) Spesifikasi ISP yang digunakan

- 2 ISP CBN up to 20Mbps

Desain Topologi Jaringan

Dalam hal ini peneliti membagi perancangan desain menjadi dua, yaitu:

a. Topologi *load balancing* Mikrotik



Gambar 3. Topologi *load balancing* Mikrotik

Gambar diatas merupakan sebagian kecil topologi jaringan yang digunakan oleh Filosofi Net. Penambahan MikroTik-1 setelah ISP1 dan ISP2 yang digunakan untuk *load balancing* kemudian diarahkan ke MikroTik-2 yang kemudian akan digunakan oleh client di Filosofi Net.

b. Topologi *load balancing* OpenWrt



Gambar 4. Topologi *load balancing* OpenWrt

Gambar diatas merupakan sebagian kecil topologi jaringan yang digunakan oleh Filosofi Net. Penambahan MikroTik-1 setelah ISP1 dan ISP2 yang digunakan untuk *load balancing* kemudian diarahkan ke MikroTik-2 yang

kemudian akan digunakan oleh client di Filosofi Net.

Hasil Pengujian

Ada dua skenario pengujian yang dilakukan pada pengujian ini. Skenario yang pertama adalah pengujian menggunakan *Internet Speed Tester*, skenario ini berfungsi untuk menguji koneksi dan kecepatan dari server. Skenario yang kedua adalah pengujian dengan melakukan *download file* sebesar 100Mb kemudian di *monitoring* menggunakan Wireshark, skenario ini berfungsi menguji *throughput*, *packet loss*, *delay* dan *jitter* sesuai dengan parameter QoS TIPHON. Berikut hasil pengujiannya:

a. Pengujian *Internet Speed Tester*

Disini peneliti melakukan pengujian dengan speedtest.cbn.id. Alasan peneliti menggunakan speedtest.cbn.id dikarenakan kedua ISP yang dipakai dalam pengujian ini berasal dari server yang sama, yaitu CBN. Berikut hasilnya:

Tabel 1. Hasil uji routerOS MikroTik

Uji	Ping	Jitter	Download	Upload
Uji ke-1	3 ms	0 ms	39.3 Mbps	38.1 Mbps
Uji ke-2	3 ms	0 ms	39.1 Mbps	38.4 Mbps
Uji ke-3	2 ms	0 ms	39.2 Mbps	38.3 Mbps
Uji ke-4	10 ms	1 ms	39.3 Mbps	38.3 Mbps
Uji ke-5	2 ms	1 ms	39.2 Mbps	38.3 Mbps
Rata-Rata	4 ms	0.4 ms	39.22 Mbps	38.28 Mbps

Berdasarkan tabel 1 diatas, maka dapat disimpulkan bahwa hasil dari pengujian ke-1 sampai dengan pengujian ke-5 tidak mengalami perubahan yang signifikan. Sehingga hasil pengujian ini mendapatkan rata-rata nilai *ping* 4 ms, *jitter* 0.4 ms, *download* 39.22 Mbps dan *Upload* 38.28 Mbps.

Tabel 2. Hasil uji OpenWrt

Uji	Ping	Jitter	Download	Upload
Uji ke-1	6 ms	3 ms	39.3 Mbps	38.3 Mbps
Uji ke-2	2 ms	1 ms	39.2 Mbps	38.4 Mbps
Uji ke-3	3 ms	1 ms	39.2 Mbps	38.3 Mbps

Uji	Ping	Jitter	Download	Upload
Uji ke-4	3 ms	1 ms	39.3 Mbps	38.2 Mbps
Uji ke-5	3 ms	1 ms	39.3 Mbps	38.3 Mbps
Rata-Rata	3.4 ms	1.4 ms	39.26 Mbps	38.3 Mbps

Berdasarkan tabel 2 diatas, maka dapat disimpulkan bahwa hasil dari pengujian ke-1 sampai dengan pengujian ke-5 tidak mengalami perubahan yang signifikan. Sehingga hasil pengujian ini mendapatkan rata-rata nilai *ping* 3.4 ms, *jitter* 1.4 ms, *download* 39.26 Mbps dan *Upload* 38.3 Mbps.

b. Pengujian Wireshark

Pada pengujian ini, peneliti menguji dengan men-download file sebesar 100 Mb. Setelah itu, akan dihitung berapa nilai *throughput*, *delay*, *packet loss* dan *jitter* setelah pengujian berlangsung. Perhitungan ini dilakukan berdasarkan hasil dari *capture* paket-paket yang muncul saat melakukan pengukuran melalui *software* Wireshark. Setelah dilakukan dengan 5 kali pengujian maka didapatkan:

Tabel 3. Hasil uji Wireshark MikroTik

Parameter	Nilai	Indeks	Kategori
<i>Throughput</i>	45.54 %	2	Sedang
<i>Packet Loss</i>	0.002 %	4	Sangat Bagus
<i>Delay</i>	0.45 ms	4	Sangat Bagus
<i>Jitter</i>	0 ms	4	Sangat Bagus

Pada tabel 3 diatas didapatkan dengan pengujian kualitas jaringan *load balancing* dengan *download file* 100 Mb di *monitoring* dengan menggunakan Wireshark. Dapat dilihat kualitas jaringan *load balancing routerOS* MikroTik mendapatkan total indeks 14.

Tabel 4. Hasil uji Wireshark OpenWrt

Parameter	Nilai	Indeks	Kategori
<i>Throughput</i>	47.51 %	2	Sedang
<i>Packet Loss</i>	0.00158 %	4	Sangat Bagus
<i>Delay</i>	0.426 ms	4	Sangat Bagus
<i>Jitter</i>	0 ms	4	Sangat Bagus

Pada tabel 4 diatas didapatkan dengan pengujian kualitas jaringan *load balancing* dengan *download file* 100 Mb di *monitoring* menggunakan Wireshark. Dapat dilihat kualitas jaringan *load balancing* OpenWrt mendapatkan total indeks 14.

4. PEMBAHASAN

Pada bagian ini, penulis membandingkan hasil pengujian berdasarkan dua skenario. Berikut hasil perbandingannya:

- a. Perbandingan hasil pengujian *Internet Speed Tester*

Tabel 5. Perbandingan uji *Internet speed Tester*

Router	Ping	Jitter	Upload	Download
MikroTik	4 ms	0.4 ms	39.22 Mbps	38.28 Mbps
OpenWrt	3.4 ms	1.4 ms	39.26 Mbps	38.3 Mbps

Berdasarkan tabel 5, maka dapat diambil kesimpulan:

1. Penulis berhasil mengkonfigurasi jaringan *load balancing* dengan menggunakan *routerOS* MikroTik dan OpenWrt. Hal ini dapat dilihat dengan *bandwidth Upload* dan *download* yang hampir mencapai 40 Mbps, ini sesuai dengan total *bandwidth* yang tersedia, yaitu 40 Mbps.
 2. *RouterOS* MikroTik memiliki jitter yang lebih rendah dibanding OpenWrt, sedangkan, OpenWrt memiliki *ping* yang lebih rendah dengan *bandwidth Upload* dan *download* yang lebih tinggi dibanding *routerOS* MikroTik.
- b. Perbandingan hasil pengujian wireshark

Tabel 6. Perbandingan uji wireshark

Router	Troughput	Packet Loss	Delay	Jitter
MikroTik	45.54 %	0.002 %	0.45 ms	0 ms
OpenWrt	47.51 %	0.00158 %	0.426 ms	0 ms

Berdasarkan tabel 6, maka dapat diambil kesimpulan:

- a. Penulis berhasil mengkonfigurasi jaringan *load balancing* dengan menggunakan *routerOS* MikroTik dan OpenWrt.

- b. Berdasarkan standar TIPHON OpenWrt memiliki kinerja yang lebih baik dibandingkan dengan *routerOS* MikroTik, terbukti pada OpenWrt memiliki *throughput* yang lebih tinggi dengan *packet loss*, *delay* dan *jitter* yang lebih rendah.

5. KESIMPULAN

Berdasarkan hasil dari penelitian yang telah dilakukan, dapat disimpulkan bahwa:

- a. Teknik *load balancing* dengan *Per Connection Classifier* pada *routerOS* MikroTik dan OpenWrt dapat berfungsi dengan baik.
- b. Kinerja *load balancing* pada pengujian dengan skenario *Internet Speed Tester*, metode PCC pada OpenWrt lebih baik dari kinerja *load balancing* metode PCC pada *routerOS* MikroTik berdasarkan perhitungan *ping*, *upload* dan *download*. Akan tetapi, nilai *jitter* pada *routerOS* MikroTik lebih baik dibanding dengan nilai *jitter* pada OpenWrt.
- c. Berdasarkan standar TIPHON pada pengujian dengan skenario *download file* 100 Mb. *RouterOS* MikroTik dan OpenWrt memiliki hasil yang sama, yaitu memiliki nilai *throughput* berkategori "sedang" dengan indeks 2 dan memiliki nilai *packet loss*, *delay* dan *jitter* berkategori "sangat bagus" dengan indeks 4.

6. DAFTAR PUSTAKA

- [1] H. Zhahir and M. Ardiansyah, "IMPLEMENTASI MANAJEMEN BANDWIDTH DENGAN MENGGUNAKAN METODE LOAD BALANCING DAN FAILOVER PADA ROUTER MIKROTIK DAN SWITCH CISCO (STUDI KASUS: PUSAT PENGUATAN DAN PEMBERDAYAAN BAHASA KEMDIKBUD RISTEK)," *INFORMATIKA*, vol. Vol.3, No.2, 2022.
- [2] A. Mikola and A. C. Nurcahyo, "Analisis Load Balancing Berbasis Mikrotik Dalam Meningkatkan Kemampuan Server di Institut Shanti Bhuana," *JIFOTECH (JOURNAL OF INFORMATION TECHNOLOGY)*, vol. 2, no. 2, 2022.
- [3] H. Kurniawan, J. Dedy Irawan, and F. X. Ariwibisono, "IMPLEMENTASI SQUID PROXY PADA MIKROTIK DAN MONITORING TRAFFIC JARINGAN BERBASIS WEBSITE," 2020.
- [4] A. N. Septandy and F. A. Fauzi, "[REVISI] Implementasi Load Balancing Menggunakan Apache Load Balancer Sejarah Web Service dan Contoh Penggunaannya View project Xampp

- mod security View project,” 2020. [Online]. Available:
<https://www.researchgate.net/publication/338372395>
- [5] D. Leman, “LOAD BALANCING 2 JALUR INTERNET MENGGUNAKAN MIKROTIK ROUND ROBIN,” 2019.
- [6] A. Salim, G. Surono, E. B. Pabelan, and A. Raizaldi, “Load Balancing Metode Per Connection Classifier Berbasis Router Mikrotik di PT.Asuransi Jiwa Nasional,” 2023. [Online]. Available:
<https://ejournal.kreatifcemerlang.id/index.php/jbpi>
- [7] R. Aldori, S. Ati, and M. Raharjo, “Implementasi Load Balancing Menggunakan Metode PCC (Per Connection Classifier) Berbasis Mikrotik pada SMK Tunas Harapan Jakarta,” *TECHSI - Jurnal Teknik Informatika*, vol. 13, no. 2, p. 69, Oct. 2021, doi: 10.29103/techsi.v13i2.5380.
- [8] F. D. Anggoro, “IMPLEMENTASI LOAD BALANCING MENGGUNAKAN RASPBERRY PI DENGAN ROUTEROS OPENWRT,” *JURNAL TRANSIT*, 2021.