

IMPLEMENTASI *RANDOM EARLY DETECTION* PADA JARINGAN WIRELESS UNTUK MENINGKATKAN *QUALITY OF SERVICE*

Afla Nevrisa¹, Siti Nadifa², Anwar³, Buyung Solihin Hasugian⁴

1,2,3) Teknologi Rekayasa Komputer Jaringan, Teknologi Informasi dan Komputer, Politeknik Negeri Lhokseumawe, Indonesia

4) Sistem Informasi, Teknik Dan Ilmu Komputer, Universitas Dharmawangsa, Indonesia

Article Info

Article history:

Received: 26 April 2026

Revised: 27 April 2026

Accepted: 30 April 2026

ABSTRACT

Abstrak

Peningkatan penggunaan jaringan nirkabel menimbulkan tantangan dalam menjaga kualitas layanan (QoS), terutama saat kepadatan trafik menyebabkan delay, jitter, packet loss, dan penurunan throughput. Penelitian ini bertujuan mengevaluasi efektivitas metode Random Early Detection (RED) dalam meningkatkan QoS pada jaringan wireless dengan beban tinggi. Pengujian dilakukan melalui enam skenario aktivitas klien, yaitu speedtest, download, upload, serta streaming YouTube resolusi standar dan 4K. Perangkat yang digunakan meliputi MikroTik, Access Point TP-Link, tiga laptop, dan dua HP, dengan pemantauan menggunakan Wireshark. Parameter QoS yang dianalisis mencakup delay, jitter, throughput, dan packet loss. Hasil menunjukkan bahwa RED mampu menurunkan delay dan jitter secara signifikan, seperti pada skenario download dari 372,14 ms menjadi 50,71 ms dan jitter dari 349,14 ms menjadi 50,75 ms. Meskipun throughput sedikit menurun akibat mekanisme packet drop, kestabilan jaringan tetap terjaga dengan packet loss 0% di seluruh skenario. Selain itu, RED mampu melakukan packet drop secara selektif saat antrian melebihi ambang batas. Dengan demikian, RED efektif sebagai solusi manajemen trafik untuk meningkatkan kinerja jaringan wireless pada kondisi padat dan multi-client aktif.

Kata Kunci: Delay, Jitter, Quality of Service, Random Early Detection, wireless

Abstract

The increasing use of wireless networks presents challenges in maintaining Quality of Service (QoS), especially under heavy traffic conditions that can lead to delay, jitter, packet loss, and reduced throughput. This study aims to evaluate the effectiveness of the Random Early Detection (RED) method in improving QoS in wireless networks with high traffic loads. The experiment was conducted using six client activity scenarios: speed testing, downloading, uploading, and YouTube streaming at standard and 4K resolutions. The setup involved MikroTik devices, a TP-Link Access Point, three laptops, and two smartphones as clients, with network monitoring performed using Wireshark. The QoS parameters analyzed include delay, jitter, throughput, and packet loss. The results show that RED significantly reduces delay and jitter. For instance, in the download scenario, delay decreased from 372.14 ms to 50.71 ms, while jitter dropped from 349.14 ms to 50.75 ms. Although throughput slightly decreased due to the packet drop mechanism, overall network stability was maintained, with packet loss consistently recorded at 0% across all scenarios. Additionally, RED effectively performs selective packet dropping when queue thresholds are exceeded. Therefore, RED can be recommended as an effective traffic management solution to enhance wireless network performance, particularly under congested conditions with multiple active clients.

Keywords: Delay, Jitter, Quality of Service, Random Early Detection, wireless

Djtechno: Jurnal Teknologi Informasi oleh Universitas Dharmawangsa Artikel ini bersifat open access yang didistribusikan di bawah syarat dan ketentuan dengan Lisensi Internasional Creative Commons Attribution NonCommercial ShareAlike 4.0 ([CC-BY-NC-SA](https://creativecommons.org/licenses/by-nc-sa/4.0/)).



1. PENDAHULUAN

Jaringan nirkabel (*wireless*) telah menjadi infrastruktur penting dalam mendukung aktivitas masyarakat modern. Penggunaannya yang luas di berbagai lingkungan didukung oleh fleksibilitas akses tanpa kabel serta kemudahan dalam mendukung berbagai layanan digital, seperti komunikasi, transaksi daring, dan komputasi berbasis cloud. Seiring meningkatnya penggunaan aplikasi real-time dan kebutuhan bandwidth besar, tuntutan terhadap koneksi yang cepat dan stabil juga semakin tinggi [1].

Namun, peningkatan jumlah pengguna dan perangkat yang terhubung sering kali menyebabkan penurunan kualitas layanan (*Quality of Service*), yang ditandai dengan meningkatnya *delay*, *packet loss*, serta ketidakstabilan *throughput*. Kondisi ini semakin kompleks akibat adanya trafik heterogen seperti multimedia, *IoT*, dan layanan *cloud*, yang berpotensi menimbulkan kemacetan jaringan jika tidak dikelola dengan baik [2][3].

Pengelolaan lalu lintas data menjadi kunci dalam menjaga performa jaringan. Metode konvensional seperti drop tail cenderung reaktif, sedangkan *Random Early Detection* (RED) menawarkan pendekatan proaktif dengan mendeteksi potensi kemacetan lebih awal. Pendekatan *Active Queue Management* seperti RED terbukti mampu meningkatkan efisiensi bandwidth serta menekan delay dan packet loss [4][5].

Penelitian ini bertujuan untuk mengimplementasikan metode RED dalam meningkatkan QoS pada jaringan nirkabel. Evaluasi dilakukan menggunakan parameter *latensi*, *jitter*, *throughput*, dan *packet loss* dengan bantuan Wireshark. Hasil penelitian diharapkan dapat memberikan kontribusi dalam pengelolaan jaringan yang lebih optimal, khususnya bagi penyedia layanan internet (ISP) [6].

2. METODE PENELITIAN

2.1 Metode dan Variabel Penelitian

Penelitian ini dilakukan dengan penelitian kuantitatif terapan yang berfokus pada pengembangan sistem. Tujuannya adalah untuk memperbaiki kualitas layanan QoS pada jaringan *wireless* melalui penerapan metode RED. Kegiatan penelitian difokuskan pada bagaimana sistem jaringan dirancang, diterapkan, serta diuji setelah menggunakan konfigurasi RED. Pendekatan ini sejalan dengan penelitian terkini yang menunjukkan

bahwa implementasi Active Queue Management seperti RED efektif dalam mengontrol kemacetan jaringan dan meningkatkan performa QoS pada jaringan modern [8].

Dalam penelitian ini, variabel yang digunakan terdiri dari:

1. Variabel bebas (Independen)

Penelitian ini adalah metode *Random Early Detection* (RED) yang diterapkan pada jaringan *wireless*. Metode RED digunakan sebagai teknik manajemen antrian yang bertujuan untuk mengurangi terjadinya kemacetan (*congestion*) pada jaringan dengan cara membuang paket secara probabilistik sebelum antrian penuh, yang terbukti efektif dalam berbagai studi jaringan terkini [9].

2. Variabel terikat (Dependen)

Quality of Service (QoS) pada jaringan *wireless* yang diukur berdasarkan beberapa parameter utama, yaitu *delay* (waktu tunda), *packet loss* (kehilangan paket), *throughput* (kecepatan aliran data), dan *jitter* (variasi waktu tunda). Keempat parameter tersebut akan dianalisis untuk melihat sejauh mana implementasi metode RED dapat meningkatkan performa jaringan, sebagaimana digunakan dalam penelitian QoS berbasis trafik dinamis [10].

2.2 Data dan Pengumpulan Data

Data penelitian diperoleh dari dua sumber, yakni data primer dan sekunder. Data primer dikumpulkan melalui eksperimen pada simulasi jaringan komputer menggunakan Mikrotik, Winbox, dan Wireshark. Mikrotik berfungsi mengatur lalu lintas, Winbox memantau jaringan secara real-time, sedangkan Wireshark menganalisis paket data. Pendekatan ini sejalan dengan metode pengujian jaringan berbasis simulasi dan monitoring paket yang banyak digunakan dalam penelitian jaringan terkini [11]. Eksperimen diawali dengan perancangan simulasi yang merepresentasikan kondisi nyata, pengaturan parameter uji (*bandwidth*, prioritas, antrian), serta penerapan metode *Random Early Detection* (RED) pada router. Selama uji, dilakukan pengukuran parameter *Quality of Service* (QoS) berupa *delay*, *jitter*, *throughput*, dan *packet loss* pada berbagai kondisi beban jaringan, dari ringan hingga padat. Hasilnya kemudian dibandingkan antara skenario dengan dan tanpa penerapan RED untuk menilai efektivitasnya [9].

Data sekunder diperoleh dari literatur terkait manajemen *bandwidth*, prinsip dan penerapan RED, serta artikel mengenai QoS. Jurnal ilmiah juga digunakan sebagai referensi untuk memahami hubungan QoS dengan performa jaringan sekaligus sebagai

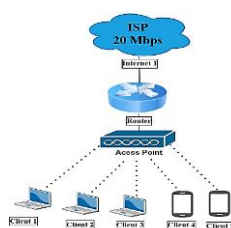
pembandingan hasil eksperimen. Dengan demikian, kombinasi data primer yang bersifat eksperimental dan sekunder yang bersifat teoretis memberikan dasar kuat bagi analisis penelitian ini [11].

2.3 Rancangan Sistem (*Hardware*)

Dalam penelitian yang berjudul *Implementasi Metode Random Early Detection Pada Jaringan Wireless Untuk Meningkatkan Quality of Service*. Dilakukan kombinasi perangkat keras dan perangkat lunak yang saling terhubung dalam satu jaringan uji. Internet dari ISP dialirkan melalui Mikrotik Routerboard sebagai router utama yang menjadi titik penerapan metode RED, kemudian didistribusikan menggunakan Access Point TP-Link TL-WA901ND agar seluruh perangkat klien dapat terhubung melalui Wi-Fi dengan SSID yang sama. Penggunaan router sebagai pusat manajemen trafik serta access point sebagai distribusi jaringan nirkabel merupakan pendekatan umum dalam pengujian QoS berbasis wireless modern[12]. Proses konfigurasi dilakukan menggunakan aplikasi Winbox, sementara analisis lalu lintas data dilakukan dengan Wireshark, keduanya dijalankan pada laptop *monitoring* yang terhubung ke jaringan. Lingkungan uji melibatkan beberapa perangkat klien yang masing-masing dilengkapi RAM 8–16 GB dan sistem operasi Windows 10 maupun Windows 11, serta dua perangkat seluler berbasis Qualcomm Snapdragon dengan RAM 8 GB dan sistem operasi ColorOS. Seluruh perangkat tersebut terhubung melalui jaringan nirkabel yang sama, sehingga membentuk lingkungan simulasi yang representatif untuk menguji efektivitas metode RED pada jaringan *wireless* untuk meningkatkan *Quality of Service*.

2.3.1 Topologi Jaringan

Untuk memahami skema koneksi jaringan dalam penelitian ini, dibuat topologi sebagai representasi alur distribusi data dari penyedia layanan internet (ISP) hingga perangkat klien. Router berfungsi mengatur lalu lintas data, sedangkan Access Point menjadi penghubung antara router dan perangkat klien. Topologi jaringan tersebut ditunjukkan pada Gambar 1.

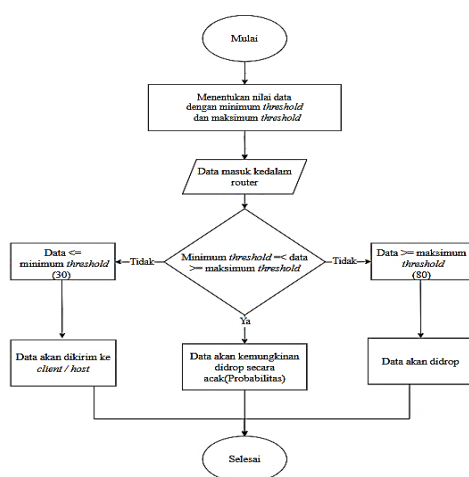


Gambar 1. Topologi Jaringan

Berdasarkan gambar diatas topologi jaringan dirancang terstruktur untuk mengimplementasikan metode RED dalam meningkatkan QoS pada jaringan wireless. Setiap komponen memiliki fungsi saling melengkapi agar pengelolaan lalu lintas data berjalan optimal. ISP menyediakan koneksi internet yang diteruskan ke router sebagai pusat distribusi sekaligus inti penerapan RED. Mekanisme RED bekerja dengan ambang batas: paket diteruskan jika di bawah minimum, dibuang jika melebihi maksimum, dan diproses probabilistik bila berada di antara keduanya, sehingga stabilitas jaringan tetap terjaga. Access Point (AP) berperan menghubungkan banyak client secara nirkabel, menghemat port router, serta mempermudah distribusi. Sementara itu, client/host menerima koneksi melalui AP dan bertindak sebagai pengirim maupun penerima data, yang selanjutnya dikelola router menggunakan RED untuk mengukur efektivitas jaringan.

2.3.2 Tahapan Penelitian

Sistematis penelitian diperjelas dengan sebuah *Flowchart* yang menunjukkan alur pengolahan data berdasarkan parameter *threshold*. Dalam hal ini administrator jaringan menentukan nilai minimum dan maksimum *threshold* sebagai acuan utama untuk proses pengiriman atau pembuangan data. *Flowchart* ini menjelaskan keputusan sistem apakah data akan dikirim, di-*drop* acak, atau di-*drop* sepenuhnya berdasarkan *threshold*. Alur penelitian lengkap ditampilkan pada Gambar 2



Gambar 2. *Flowchart* Tahapan Penelitian

Berikut penjelasan langkah-langkah dalam *Flowchart* tersebut:

1. Mulai: Menentukan parameter ambang batas, yaitu minimum *threshold* (30) dan maximum *threshold* (80).

2. Data Masuk: Data diperiksa apakah nilainya di bawah, di antara, atau melebihi *threshold*.
3. Proses di Router: Router mengelola lalu lintas data sesuai nilai ambang batas.
 - a. Jika data ≤ 30 → diteruskan ke *client/host* tanpa pembuangan.
 - b. Jika $30 < \text{data} < 80$ → diproses dengan probabilitas tertentu, dapat diteruskan atau dibuang secara acak.
 - c. Jika data ≥ 80 → otomatis dibuang (*drop*).
4. Output: Data yang diteruskan sampai ke *client/host*, sedangkan data yang dibuang tidak diteruskan sehingga stabilitas jaringan tetap terjaga.
5. Selesai: Proses berakhir setelah semua data diproses sesuai kondisi yang ditetapkan.

2.4 Teknik Pengujian

Pada skenario pertama, seluruh client menjalankan speedtest bersamaan untuk mengetahui kapasitas bandwidth dasar (4 Mbps) sebelum dan sesudah penerapan RED. Skenario kedua, semua client mengunduh file ISO besar (Kali Linux dan Debian netinst) secara serentak guna memaksimalkan trafik masuk dan menilai kinerja jaringan pada beban tinggi berdasarkan *throughput*, *delay*, *jitter*, dan *packet loss*. Skenario ketiga menguji aktivitas upload file ke Google Drive untuk menilai efektivitas RED dalam mengelola antrian data keluar. Pada skenario keempat, tiap client melakukan aktivitas berbeda: browsing, download ISO, dan streaming YouTube (480p–720p). Kondisi ini merepresentasikan campuran trafik ringan, sedang, dan berat, termasuk trafik sensitif terhadap delay.

Skenario kelima melibatkan seluruh client menonton YouTube pada kualitas standar untuk menguji kestabilan RED dalam menangani trafik multimedia. Sedangkan skenario keenam menguji streaming YouTube beresolusi 4K secara bersamaan, menghasilkan trafik konstan dengan kebutuhan bandwidth tinggi. Fokus utamanya adalah *delay* dan *jitter* untuk mengevaluasi stabilitas QoS pada kondisi padat. Hasil dari keenam skenario dibandingkan dengan standar QoS untuk memperoleh gambaran menyeluruh mengenai efektivitas RED dalam mengelola lalu lintas data pada berbagai kondisi jaringan.

3. HASIL DAN PEMBAHASAN

Penelitian ini dilakukan untuk mengetahui seberapa efektif penerapan metode RED pada jaringan *wireless* dalam upaya meningkatkan kualitas layanan (*Quality of Service/QoS*).

3.1 Hasil Penelitian

Pengujian dilakukan dengan membandingkan performa jaringan sebelum dan sesudah penerapan metode RED. Parameter QoS yang diukur mencakup *throughput*, *delay*, *jitter*, dan *packet loss*, yang semuanya merupakan indikator penting dalam menilai performa layanan jaringan. Proses pengukuran dilakukan menggunakan perangkat lunak *Wireshark*, yang mampu merekam dan menganalisis data lalu lintas jaringan secara menyeluruh. Pengujian dilakukan dalam enam skenario berbeda yang merepresentasikan aktivitas pengguna jaringan secara nyata.

3.1.1 Lima *Client* melakukan *speedtest*

Pada tahap pengujian ini, dilakukan uji coba dengan melibatkan lima *client* yang menjalankan *speedtest* di waktu yang sama. Kondisi jaringan saat itu dipastikan stabil, dengan *bandwidth* total sebesar 20 Mbps yang dibagikan secara merata oleh ISP ke semua *client*. Artinya, masing-masing *client* mendapatkan alokasi *bandwidth* sebesar 4 Mbps. Karena pembagian *bandwidth* sudah seimbang sejak awal, tidak terlihat adanya perbedaan yang signifikan antara kondisi jaringan tanpa menggunakan metode RED dan saat metode RED diterapkan. Hasil pengujian *speedtest* pada *client* dapat dilihat seperti pada gambar 3 berikut.



Gambar 3. Hasil Pengujian *Speedtest*

3.1.2 Lima *Client* melakukan aktivitas *download* secara bersamaan.

Pada skenario ini, kelima *client* melakukan pengunduhan dua file ISO berukuran besar secara bersamaan untuk mensimulasikan kondisi jaringan dengan beban trafik yang tinggi. File yang diunduh berasal dari tautan:

- a. <https://xsrv.moratelindo.io/kali-images/kali-2025.2/kali-linux-2025.2-installer-amd64.iso>
- b. <https://laotzu.ftp.acc.umu.se/debian-cd/current/amd64/iso-cd/debian-12.11.0-amd64-netinst.iso>

Tujuan dari pengujian ini adalah untuk melihat bagaimana performa jaringan saat seluruh *client* aktif mengakses data secara bersamaan. Parameter yang diamati meliputi *throughput*, *packet loss*, *delay*, *jitter*, dan *packet loss*, dan dapat dilihat dari Tabel 1 dan Tabel 2 berikut.

Tabel 1. Skenario 2 Tanpa Metode RED

Kategori	Tanpa RED Skenario 2		
	Download	Indeks	Kategori
<i>Troughput</i>	2 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	372.14 ms	2	Sedang
<i>Jitter</i>	349.14 ms	1	Tidak bagus
<i>Ping Average</i>	171.8	3	Bagus

Tabel 2. Skenario 2 Metode RED

Kategori	RED Skenario 2		
	Download	Indeks	Kategori
<i>Troughput</i>	102 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	50.71 ms	4	Sangat Bagus
<i>Jitter</i>	50.75ms	3	Bagus
<i>Ping Average</i>	64.2	4	Sangat Bagus

Dari Tabel 1 dan Tabel 2 terlihat bahwa penggunaan RED memberikan hasil lebih baik dibandingkan tanpa RED. Meskipun *throughput* masih rendah metode ini mampu menurunkan *delay* serta *jitter*. dan nilai ping juga membaik. Sementara *packet loss* tetap masuk kategori sangat baik, secara keseluruhan, RED meningkatkan stabilitas jaringan terutama pada parameter *delay*, *jitter*, dan ping.

3.1.3 Lima *Client* melakukan aktivitas *upload* secara bersamaan.

Pada skenario ini, seluruh *client* melakukan unggah file secara bersamaan ke layanan seperti Google Drive. Tujuannya adalah untuk melihat bagaimana metode RED

bekerja saat jaringan mengalami beban pada trafik keluar, serta dampaknya terhadap parameter QoS yang diamat, untuk melihat hasil dari penerapan tanpa RED dan metode RED dapat dilihat dari Tabel 3 dan Tabel 4 berikut.

Tabel 3. Skenario 3 Tanpa Metode RED

Kategori	Tanpa RED Skenario 3		
	Upload	Indeks	Kategori
<i>Troughput</i>	5 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	284.16 ms	3	Bagus
<i>Jitter</i>	285.24 ms	1	Tidak bagus
<i>Ping Average</i>	137.4	4	Sangat Bagus

Tabel 4. Skenario 3 Metode RED

Kategori	RED Skenario 3		
	Upload	Indeks	Kategori
<i>Troughput</i>	36 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	112.96 ms	4	Sangat Bagus
<i>Jitter</i>	113.34 ms	2	Sedang
<i>Ping Average</i>	64.6	4	Sangat Bagus

3.1.4 Satu *client* melakukan *browsing*, dua *client* melakukan *download*, dan dua *client* menonton video YouTube.

Pada skenario ini, setiap perangkat *client* menjalankan aktivitas yang berbeda secara bersamaan untuk menguji kestabilan jaringan dengan kombinasi trafik kecil, sedang, dan tinggi:

- Satu perangkat digunakan untuk *browsing* di <https://www.apple.com/id/>,
- Dua perangkat digunakan untuk mengunduh file dari <https://xsrv.moratelindo.io/kali-images/kali-2025.2/kali-linux-2025.2-installer-amd64.iso>,
- Dua perangkat lainnya digunakan untuk menonton YouTube dengan kualitas 480p–720p.

Untuk melihat hasil dari penerapan tanpa RED dan metode RED dapat dilihat dari Tabel 5 dan Tabel 6 berikut.

Tabel 5. Skenario 4 Tanpa Metode RED

Kategori	Tanpa RED Skenario 4		
	1 Client Browsing 2,3 Download 4,5 Youtube	Indeks	Kategori
<i>Troughput</i>	52 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	67.37 ms	4	Sangat Bagus
<i>Jitter</i>	67.55 ms	3	Bagus
<i>Ping Average</i>	113.4	4	Sangat Bagus

Tabel 6. Skenario 4 Metode RED

Kategori	RED Skenario 4		
	1 Laptop Browsing 2,3 Download 4,5 Youtube	Indeks	Kategori
<i>Troughput</i>	150 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	21.06 ms	4	Sangat Bagus
<i>Jitter</i>	22.03 ms	3	Bagus
<i>Ping Average</i>	64.6	4	Sangat Bagus

3.1.5 Lima Client menonton video YouTube dengan kualitas biasa.

Pada skenario ini, seluruh perangkat *client* digunakan untuk menonton video di YouTube dengan kualitas standar (480p–720p). Tujuannya adalah untuk menguji performa metode RED dalam menangani trafik multimedia ringan hingga sedang, serta memastikan layanan tetap stabil dan sesuai standar QoS. Untuk melihat hasil dari penerapan tanpa RED dan metode RED dapat dilihat dari Tabel 7 dan Tabel 8 berikut.

Tabel 7. Skenario 5 Tanpa Metode RED

Kategori	Tanpa RED Skenario 5		
	Menonton Youtube	Indeks	Kategori

<i>Throughput</i>	14 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	161.47 ms	3	Bagus
<i>Jitter</i>	161.85 ms	1	Tidak bagus
<i>Ping Average</i>	79.8	4	Sangat Bagus

Tabel 8. Skenario 5 Metode RED

Kategori	RED Skenario 5		
	Menonton Youtube	Indeks	Kategori
<i>Throughput</i>	13 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	168.47 ms	3	Bagus
<i>Jitter</i>	167.37 ms	1	Tidak bagus
<i>Ping Average</i>	66.6	4	Sangat Bagus

Dari kedua table diatas, tanpa metode RED *throughput* yang tercatat masuk kategori buruk. *Packet loss* sangat bagus karena tidak ada data yang hilang sedangkan *Delay* dan *jitter* masing-masing berada pada kategori bagus dan tidak bagus. Ping termasuk dalam kategori sangat bagus. Setelah penerapan metode RED, hasilnya tidak menunjukkan adanya peningkatan signifikan. *Throughput* justru menurun dan tetap berada dalam kategori buruk. *Packet loss* tetap dengan kategori sangat bagus. *Delay* sedikit meningkat namun masih dalam kategori yang sama, yaitu bagus. *Jitter* juga mengalami kenaikan dan tetap dalam kategori tidak bagus. Sementara itu, ping average turun namun tidak mengubah kategorinya, yaitu sangat bagus. Secara keseluruhan, penerapan metode RED pada skenario ini tidak memberikan perbaikan berarti terhadap parameter jaringan, baik dari sisi nilai maupun kategori indeksinya.

3.1.6 Lima Client menonton video YouTube dengan kualitas 4K.

Pada skenario ini, semua *client* menonton video YouTube 4K secara bersamaan melalui tautan <https://www.youtube.com/watch?v=nKK9d09fFSyc&t=6137s> dengan *bandwidth* dibatasi 4 Mbps per *client*. Aktivitas ini menciptakan trafik konstan dan berat, sehingga fokus pengujian diarahkan pada *jitter* dan *delay* untuk menilai kestabilan QoS. Hasilnya kemudian dibandingkan dengan standar QoS untuk menentukan apakah jaringan sudah memenuhi kriteria performa yang baik. Untuk

melihat hasil dari penerapan tanpa RED dan metode RED dapat dilihat dari Tabel 9 dan Tabel 10 berikut

Tabel 9. Skenario 6 Tanpa Metode RED

Kategori	Tanpa RED Skenario 6		
	Menonton Youtube 4K	Indek	Kategori
<i>Troughput</i>	7 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	238.16 ms	3	Bagus
<i>Jitter</i>	236.36 ms	1	Tidak Bagus
<i>Ping Average</i>	93.8	4	Sangat Bagus

Tabel 10. Skenario 6 Metode RED

Kategori	RED Skenario 6		
	Menonton Youtube 4K	Indeks	Kategori
<i>Troughput</i>	14 Kbps	0	Buruk
<i>Paket Loss</i>	0%	4	Sangat Bagus
<i>Delay</i>	50.74 ms	4	Sangat Bagus
<i>Jitter</i>	50.77 ms	3	Bagus
<i>Ping Average</i>	78	4	Sangat Bagus

Dari masing- masing skenario di pengujian dilakukan dua kali pada masing-masing skenario, yaitu tanpa dan dengan penerapan metode RED. Parameter yang diuji meliputi *throughput*, *packet loss*, *delay*, *jitter*, dan *ping average*, yang kemudian dikategorikan ke dalam indeks serta klasifikasi kualitas layanan. Secara umum, penerapan RED terbukti mampu meningkatkan stabilitas jaringan, terutama dalam parameter *delay* dan *jitter*.

3.2 Pembahasan Penelitian

Adapun hasil dari penelitian yang dilakukan berdasarkan dua skenario pengujian yang telah dijelaskan sebelumnya, yaitu sebelum penerapan metode RED dan sesudah penerapannya. Pengujian dapat dilihat dibawah ini.

3.2.1 Hasil Implementasi RED Dari Skenario Data yang *Didropped*

Pengujian implementasi RED dilakukan pada jaringan Wi-Fi dengan SSID "Siti", di mana seluruh *client* menjalankan skenario secara bersamaan. Pada tahap awal tanpa penerapan RED, tidak ditemukan adanya *packet drop* karena sistem antrian masih menggunakan konfigurasi *default* tanpa mekanisme kontrol kemacetan, sebagaimana

terlihat pada hasil terminal (*/queue simple print stats*) untuk *Client-1* dan *Client-3* seperti ditunjukkan pada Gambar 4 dan Gambar 5 Berikut.

```
1  name="client-1 " target=192.168.10.10/32 rate=584bps/480bps total-rate=0bps
   packet-rate=1/1 total-packet-rate=0 queued-bytes=0/0 total-queued-bytes=0
   queued-packets=0/0 total-queued-packets=0 bytes=5156530/119746298
   total-bytes=0 packets=39432/92242 total-packets=0 dropped=0/7033
   total-dropped=0
```

Gambar 4. *Client-1* Tidak Ada Data yang *Didropped*

```
3  name="client-3" target=192.168.10.13/32 rate=21.6kbps/873.2kbps
   total-rate=0bps packet-rate=31/91 total-packet-rate=0 queued-bytes=0/0
   total-queued-bytes=0 queued-packets=0/0 total-queued-packets=0
   bytes=5070738/79727914 total-bytes=0 packets=23960/68485 total-packets=0
   dropped=0/2033 total-dropped=0
```

Gambar 5. *Client-3* Tidak Ada Data yang *Didropped*

Setelah metode RED diterapkan, mulai muncul *packet drop* akibat trafik data melebihi kapasitas antrian yang telah dikonfigurasi. RED bekerja dengan membuang paket secara selektif sebelum antrian penuh, menggunakan parameter kapasitas 100 paket, ambang minimum 30 paket, ambang maksimum 80 paket, nilai *burst* 50 paket, serta ukuran rata-rata paket 1000 byte. Ketika antrian melewati ambang minimum, RED mulai membuang paket secara acak dengan intensitas yang meningkat mendekati ambang maksimum. Proses ini bertujuan untuk menghindari antrian penuh yang dapat memicu peningkatan *delay* atau *jitter* dalam jaringan. Informasi jumlah paket yang terbuang dapat dilihat melalui terminal menggunakan perintah */queue simple print stats*, yang menampilkan statistik performa secara langsung sebagaimana ditampilkan pada gambar 6 dan gambar 7 berikut.

```
1  name="client-1 " target=192.168.10.10/32 rate=45.4kbps/4.0Mbps
   total-rate=0bps packet-rate=135/346 total-packet-rate=0 queued-bytes=0/0
   total-queued-bytes=0 queued-packets=0/0 total-queued-packets=0
   bytes=4758979/170100281 total-bytes=0 packets=51199/122345
   total-packets=0 dropped=99/1631 total-dropped=0
```

Gambar 6. Hasil *Dropped* pada *Client-1*

```
3  name="client-3" target=192.168.10.13/32 rate=22.8kbps/1632.0kbps
   total-rate=0bps packet-rate=63/140 total-packet-rate=0 queued-bytes=0/0
   total-queued-bytes=0 queued-packets=0/0 total-queued-packets=0
   bytes=26324567/560630575 total-bytes=0 packets=239564/420574
   total-packets=0 dropped=110/23443 total-dropped=0
```

Gambar 7. Hasil *Dropped* pada *Client-3*

Hasil pengujian menunjukkan *Client-1* mengalami *packet drop* sebanyak 99 dari total 1.631 paket, sedangkan *Client-3* mengalami 110 *packet drop* dari total 23.443 paket. Perbedaan ini mengindikasikan adanya variasi beban trafik pada tiap *client* yang ditangani RED secara dinamis. Meskipun terjadi pembuangan paket, mekanisme ini

berfungsi menjaga kestabilan jaringan dengan mencegah antrian penuh yang berpotensi meningkatkan *delay*, *jitter*, bahkan kehilangan koneksi. Dengan demikian, penerapan RED terbukti efektif dalam mengontrol antrian secara adaptif untuk menjaga performa jaringan pada kondisi trafik tinggi.

4. SIMPULAN

Dari hasil pengujian dan analisis yang dilakukan terhadap penerapan metode *Random Early Detection* (RED) pada jaringan *wireless*, dapat disimpulkan beberapa hal sebagai berikut:

- a. Penerapan metode RED terbukti efektif meningkatkan kualitas layanan QoS, terutama dengan menurunkan *delay* dan *jitter* secara signifikan. Pada skenario *download*, *delay* turun dari 372,14 ms menjadi 50,71 ms dan *jitter* dari 349,14 ms menjadi 50,75 ms. Meski *throughput* pada beberapa skenario tanpa RED lebih tinggi, penggunaan RED memberikan perbaikan stabilitas dengan penurunan *delay* dan *jitter* yang nyata. *Packet loss* tetap 0% di semua skenario, menunjukkan integritas data terjaga. Nilai rata-rata ping juga lebih rendah dengan RED, menandakan waktu tunda rata-rata berkurang dan koneksi lebih responsif. RED terbukti mampu melakukan *packet drop* selektif saat trafik berlebih, sementara pada skenario YouTube kualitas standar tidak ada peningkatan *delay* maupun *jitter*, sedangkan pada resolusi 4K terjadi peningkatan keduanya.
- b. RED berperan penting dalam mengelola performa jaringan berdasarkan parameter QoS. Hasil pengujian menunjukkan RED mampu menjaga *delay* dan *jitter* tetap optimal dengan penurunan signifikan di hampir semua skenario. Walaupun *throughput* tidak selalu tertinggi, stabilitas dan konsistensi nilai QoS membuktikan bahwa RED efektif sebagai mekanisme manajemen trafik, mencegah kemacetan lebih awal, menjaga kualitas transmisi data, dan meningkatkan pengalaman pengguna.

REFERENCES

- [1]. M.-C. Chou, G.-J. Lin, C.-H. Ke, and Y.-S. Chen, "Differentiated QoS Provisioning in Wireless Networks Based on Deep Reinforcement Learning," *Advances in Technology Innovation*, vol. 9, no. 4, pp. 257–272, 2024.
- [2]. A. Wijaya, A. Abdullah, E. Windriyani, F. C. Samaeni, M. Y. Romdhan, R. Ardiansah, dan T. Thoyyibah, "Implementasi *Quality of Service* (QoS) menggunakan Wireshark pada Jaringan

- WLAN LAN,” *Digital Transformation Technology*, vol. 4, no. 1, hlm. 296–303, 2024. doi: 10.47709/digitech.v4i1.4030.
- [3]. A .Orman, Ç. Elmas, and İ. Güler, “Multi-Criteria Priority RED Queuing (MCPRQ): A Novel Approach to Enhance QoS in IPv6 Networks,” *Computers and Informatics*, vol. 4, no. 2, 2024.
- [4]. M. Q. Matrood and M. H. Ali, “Enhancing Network Congestion Control: A Comparative Study of Traditional and AI-Enhanced Active Queue Management Techniques,” *Journal of Cybersecurity and Information Management*, 2024.
- [5]. C. P. Bandhaso dan Y. Sutanto, “Analisis dan Implementasi Metode *Random Early Detection* (RED) pada Jaringan TCP/IP,” *Respati*, vol. 17, no. 2, hlm. 51, 2022. doi: 10.35842/jtir.v17i2.458.
- [6]. C Zhang, J Yang, N Wang, “An Active Queue Management for Wireless Sensor Networks with Priority Scheduling Strategy,” *Journal of Parallel and Distributed Computing*, vol. 187, 2024.
- [7]. H. Zikri, I. Iskandar, dan P. Pizaini, “Analisis Kualitas Jaringan Internet Kampus Universitas Islam Negeri Sultan Syarif Kasim Riau Menerapkan Metode *Quality of Service* (QoS),” *JURIKOM (Jurnal Riset Komputer)*, vol. 9, no. 5, hlm. 1502, 2022. doi: 10.30865/jurikom.v9i5.4930.
- [8]. I. S. Zaryadov et al., “Chronology of the Development of Active Queue Management Algorithms of RED Family (2016–2023),” *Discrete and Continuous Models and Applied Computational Science*, vol. 32, no. 2, 2024.
- [9]. A. Orman, Ç. Elmas, and İ. Güler, “Multi-Criteria Priority RED Queuing (MCPRQ): A Novel Approach to Enhance QoS in IPv6 Networks,” *Computers and Informatics*, vol. 4, no. 2, 2024.
- [10]. H. Mahmoud et al., “QoS Provisioning and Resource Block Management in AI-Enabled Networks,” *IEEE WCNC*, 2024.
- [11]. C Zhang, J Yang, N Wang, “An Active Queue Management for Wireless Sensor Networks with Priority Scheduling Strategy,” *Journal of Parallel and Distributed Computing*, vol. 187, 2024.
- [12]. D. I. S. Laia, D. Hia, Y. A. Zalukhu, T. Zebua, R. Kaban, dan D. Y. B. Ginting, “Implementasi *Simple Queue* pada Router Mikrotik RB941-2nd untuk meningkatkan *Quality of Service* Jaringan Internet Kampus,” *Bulletin of Computer Science Research*, vol. 5, no. 5, hlm. 1212–1223, 2025.
- [13]. I. Nurrobi, K. Kusnadi, dan R. Adam, “Penerapan Metode QoS (*Quality of Service*) untuk Menganalisa Kualitas Kinerja Jaringan *Wireless*,” *Jurnal Digit: Digital of Information Technology*, vol. 10, no. 1, hlm. 47–58, 2020.
- [14]. A. Basri and B. Yuliadi, “*Wireless Network Bandwidth Quality Measurement Using QoS Standard Tiphon*,” *PIKSEL: Penelitian Ilmu Komputer Sistem Embedded and Logic*, vol. 11, no. 2, pp. 283–292, 2023.
- [15]. A. Alzi dan H. Haeruddin, “Pengaruh manajemen *bandwidth* terhadap QoS dengan standar TIPHON pada alur *monitoring* SNMP,” *Jurnal Ilmiah Teknologi Informasi Asia*, vol. 17, no. 1, hlm. 9–20, 2023.
- [16]. Manuputty, Y. Y., & Widiyari, I. R. (2024). “*Analisis Performa Jaringan Software Defined Networking dengan Algoritma Random Early Detection*”. *Jurnal JTik*, 8(2), 324–332. DOI: 10.35870/jtik.v8i2.1745
- [17]. Abdulrehman Arif, Furqan Jamil, Syed Zohair Quain Haider, & Muhammad Zeeshan Haider Ali. (2025). “OPTIMIZING CONGESTION CONTROL FOR QUALITY OF SERVICE (QOS) IN

-
- BANDWIDTH-CONSTRAINED WIRELESS NETWORKS". *Spectrum of Engineering Sciences*, 3(6), 336–364. Retrieved from <https://thesesjournal.com/index.php/1/article/view/468>.
- [18]. M Rostami, S Goli-Bidgoli, (2024). "An overview of QoS-aware load balancing techniques in SDN-based IoT networks". *Journal of Cloud Computing* 13:89 <https://doi.org/10.1186/s13677-024-00651-7>
- [19]. PK Deshmukh, DT Mane, (2023). "QoS-Aware Routing and Resource Allocation Techniques for Enhanced Network Performance". *Journal of Electrical Systems*, 2023, Vol 19, Issue 2, p78
- [20]. A. Alzi dan H. Haeruddin, "Pengaruh manajemen *bandwidth* terhadap QoS dengan standar TIPHON pada alur *monitoring* SNMP," *Jurnal Ilmiah Teknologi Informasi Asia*, vol. 17, no. 1, hlm. 9–20, 2023.