Implementasi Sistem Monitoring Pppoe Berbasis Telegram Bot Api Menggunakan Fitur Netwatch Pada Router Mikrotik

Kodar Udoyono[1*], Mela Puspita Sari[1], Eka Permana[1], Setia Aji[1], Daud Elia Leander[1], Anderias Eko Wijaya[2], Dicky Iskandar Sobari[3]

[1]*Teknik Komputer dan Jaringan, Fakultas Teknik, Universitas Mandiri*
[2]*Teknik Informatika, Fakultas Teknik, Universitas Mandiri*
[3]*Sistem Informasi, Fakultas Teknik, Universitas Mandiri*

Email Korespodensi: kodarudoyono@gmail.com[*]

Received: 2026-08-29 | Accepted: 2026-10-29 | Published: 2026-10-30

Abstrak

Ketersediaan layanan *Internet Service Provider* (ISP) dipengaruhi oleh kecepatan operator dalam mendeteksi perubahan status koneksi pelanggan. Pada *Erlangga Network*, pemantauan yang sebelumnya bergantung pada pemeriksaan manual berpotensi menunda penanganan gangguan. Penelitian ini bertujuan mengimplementasikan sistem monitoring sesi *Point-to-Point Protocol over Ethernet* (PPPoE) berbasis Telegram Bot API menggunakan fitur *Netwatch* pada router MikroTik. Metode penelitian meliputi analisis kebutuhan, perancangan event handler pada PPP Profile, konfigurasi *Netwatch* dan Telegram Bot API, pengujian fungsional berbasis *black-box*, serta analisis paket menggunakan *Wireshark*. Pada event On Up, sistem mengambil *username* dan *Remote IP*, mengirimkan notifikasi, kemudian mendaftarkan alamat tersebut ke *Netwatch*. Pada event *On Down*, sistem mengirimkan notifikasi pemutusan dan menghapus entri pemantauan terkait. Hasil pengujian menunjukkan bahwa seluruh skenario fungsional berhasil. Pengujian kinerja mencatat delay ICMP 3,18 ms, *jitter* 1,12 ms, *packet* loss 0% pada kondisi normal, dan latensi notifikasi 1,40–1,80 detik. Implementasi ini mengotomatisasi pengelolaan target *Netwatch* berdasarkan siklus hidup sesi PPPoE dan menyediakan peringatan dini melalui perangkat seluler tim teknis.

Kata kunci: Monitoring Jaringan, *Netwatch*, PPPoE, *RouterOS*, Telegram Bot API.

Abstract

Internet Service Provider (ISP) service availability depends on how quickly operators detect changes in customer connectivity. At Erlangga Network, monitoring that previously relied on manual inspection could delay incident handling. This study implements a Point-to-Point Protocol over Ethernet (PPPoE) session monitoring system based on the Telegram Bot API and the Netwatch feature on a MikroTik router. The method comprises requirements analysis, PPP Profile event-handler design, Netwatch and Telegram Bot API configuration, black-box functional testing, and packet analysis using Wireshark. During the On Up event, the system obtains the username and Remote IP, sends a notification, and registers the address in Netwatch. During the On Down event, the system sends a disconnection notification and removes the related monitoring entry. The results show that all functional scenarios were successful. Performance testing recorded an ICMP delay of 3.18 ms, jitter of 1.12 ms, zero packet loss under normal conditions, and notification latency of 1.40–1.80 seconds. The implementation automates Netwatch target management according to the PPPoE session lifecycle and provides early warnings to the technical team via mobile devices.

Keywords*: Network Monitoring, Netwatch, PPPoE, RouterOS, Telegram Bot API.*

1. Pendahuluan

Ketersediaan infrastruktur jaringan merupakan salah satu faktor utama dalam menjaga kualitas layanan dan efisiensi operasional penyedia jasa internet. Gangguan pada router, jalur distribusi, atau perangkat akses dapat menyebabkan terhentinya layanan bagi banyak pelanggan. Karena itu, ISP memerlukan pemantauan proaktif yang mampu menyampaikan perubahan status koneksi kepada operator secara cepat [1], [7].

Erlangga Network (PT. Baraya Bareng Sadulur) mengelola layanan pelanggan berbasis PPPoE pada infrastruktur jaringan FTTH. Pemantauan konektivitas yang dilakukan secara manual atau berdasarkan keluhan pelanggan bersifat reaktif. Pola tersebut dapat memperpanjang waktu identifikasi gangguan dan menunda tindakan teknis. Sistem peringatan dini diperlukan agar perubahan status koneksi dapat diketahui tanpa ketergantungan pada akses terus-menerus ke antarmuka manajemen router.

MikroTik RouterOS menyediakan Netwatch untuk menguji ketersediaan host secara periodik melalui ICMP dan mengeksekusi skrip ketika status *host* berubah [1], [2]. Namun, alamat *IP remote* pada sesi PPPoE dialokasikan ketika pelanggan terhubung dan dapat berubah setelah koneksi ulang. Pendaftaran target *Netwatch* secara statis tidak sesuai untuk kondisi tersebut karena administrator harus memperbarui daftar host secara manual.

Penelitian ini mengatasi masalah tersebut dengan menempatkan *event-driven scrip*t pada PPP Profile. Saat sesi terbentuk, sistem mengambil username dan Remote IP untuk mengirimkan notifikasi serta menambahkan target *Netwatch*. Saat sesi berakhir, sistem mengirimkan notifikasi pemutusan dan menghapus target terkait. Telegram Bot API digunakan sebagai kanal notifikasi melalui permintaan HTTPS sehingga informasi dapat diterima oleh tim teknis pada perangkat seluler.

Rumusan masalah penelitian adalah bagaimana mengelola target *Netwatch* secara dinamis berdasarkan siklus hidup sesi PPPoE, bagaimana mengirimkan notifikasi perubahan status melalui Telegram Bot API, dan bagaimana memvalidasi fungsi sistem melalui pengujian fungsional, analisis paket, serta pengukuran kinerja. Tujuan penelitian adalah mengimplementasikan dan mengevaluasi mekanisme tersebut pada *Core Router MikroTik* Erlangga Network. Kontribusi penelitian terletak pada sinkronisasi otomatis antara sesi PPPoE dan daftar target *Netwatch* tanpa pendaftaran alamat IP manual.

1.1 Posisi penelitian dan kebaruan

Studi mengenai *Netwatch* pada *MikroTik* telah menunjukkan bahwa pengujian ICMP dan *event script* dapat digunakan untuk memantau ketersediaan *host* serta mengirimkan peringatan kepada administrator [1], [2]. Otomatisasi pengawasan infrastruktur juga dilaporkan dapat mengurangi ketergantungan pada pemeriksaan manual dan mempercepat pencatatan gangguan [3]. Penelitian mengenai PPPoE menempatkan autentikasi dan alokasi alamat sebagai bagian penting dari pengelolaan akses pelanggan [13].

Keterbatasan pendekatan berbasis host statis muncul ketika alamat target berubah mengikuti pembentukan dan pemutusan sesi PPPoE. Administrator tidak hanya perlu menerima informasi gangguan, tetapi juga menjaga konsistensi daftar host agar tidak memantau alamat yang sudah tidak digunakan. Celah tersebut membedakan penelitian ini dari pemantauan perangkat tetap karena target Netwatch dikelola berdasarkan siklus hidup sesi pelanggan.

Kebaruan penelitian terletak pada integrasi tiga fungsi dalam satu alur event-driven: PPP Profile menyediakan konteks username dan Remote IP, *Netwatch* melakukan verifikasi ICMP, dan Telegram Bot API menyampaikan perubahan status kepada tim teknis. Selain validasi fungsional, penelitian ini menyertakan inspeksi paket dan perhitungan QoS dengan definisi operasional yang eksplisit. Dengan demikian, kontribusi penelitian tidak hanya berupa notifikasi, tetapi juga sinkronisasi otomatis antara sesi PPPoE dan target pemantauan.

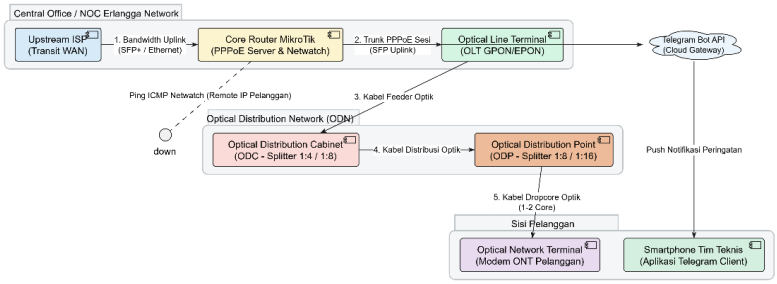
2. Metode Penelitian

2.1 Desain penelitian

Penelitian menggunakan pendekatan implementatif dengan tahapan analisis kebutuhan, perancangan arsitektur, konfigurasi sistem, pengujian fungsional, inspeksi paket, dan evaluasi kinerja. Pengujian fungsional menggunakan metode black-box dengan mengamati kesesuaian keluaran sistem terhadap kondisi sesi PPPoE. Analisis paket digunakan untuk memeriksa pertukaran ICMP yang menjadi dasar deteksi Netwatch, sedangkan pengukuran kinerja mencakup delay, packet loss, jitter, ukuran payload notifikasi, dan latensi penyampaian notifikasi.

2.2 Lingkungan penelitian

Objek penelitian adalah sesi PPPoE pelanggan pada infrastruktur Erlangga Network. *Core Router MikroTik* menjalankan PPPoE *Server*, PPP *Profile*, *Netwatch*, *RouterOS scripting*, dan fasilitas/*tool fetch*. Infrastruktur akses menggunakan FTTH berbasis GPON/EPON dengan ONT pelanggan sebagai perangkat pembentuk sesi. Komputer administrator digunakan untuk konfigurasi melalui *Winbox* dan analisis paket menggunakan *Wireshark*. Telegram Client pada perangkat seluler tim teknis digunakan untuk menerima notifikasi.



Gambar 1. Arsitektur jaringan FTTH dan integrasi kanal notifikasi

Gambar 1 memperlihatkan jalur layanan dari upstream ISP menuju *Core Router*, OLT, jaringan distribusi optik, dan ONT pelanggan. Diagram juga menunjukkan dua jalur fungsi sistem: ICMP digunakan untuk memantau keterjangkauan sesi PPPoE, sedangkan HTTPS digunakan untuk meneruskan notifikasi dari router ke Telegram Bot API dan tim teknis.

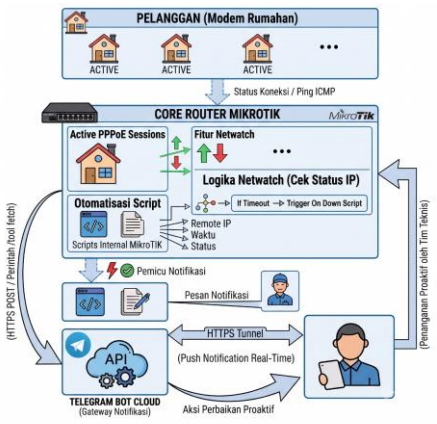
Tabel 1. Lingkungan perangkat keras dan perangkat lunak penelitian

Komponen	Spesifikasi atau versi	Fungsi
Core Router	MikroTik x86_64; RAM 32 GB; storage 128 GB; 6 SFP+	PPPoE Server, RouterOS scripting, dan Netwatch
Sistem operasi router	RouterOS 6.49.18	Menyediakan layanan jaringan dan event handler
Komputer administrator	Intel Core i3-10105F; RAM 32 GB; Windows 11 64-bit	Konfigurasi router dan analisis data
Aplikasi manajemen	Winbox 3.43 64-bit	Akses dan konfigurasi RouterOS
Perangkat pelanggan	ONT EPON/GPON	Membentuk sesi PPPoE dan menerima Remote IP
Kanal notifikasi	Telegram Bot API dan Telegram Client	Mengirim dan menerima peringatan status

Tabel 1 merangkum komponen perangkat keras dan perangkat lunak yang digunakan dalam penelitian. Spesifikasi tersebut menunjukkan bahwa *Core Router* menjalankan fungsi utama PPPoE *Server, scripting,* dan *Netwatch*, sedangkan komputer administrator dan perangkat seluler mendukung konfigurasi serta penerimaan notifikasi.

2.3 Arsitektur dan mekanisme sistem

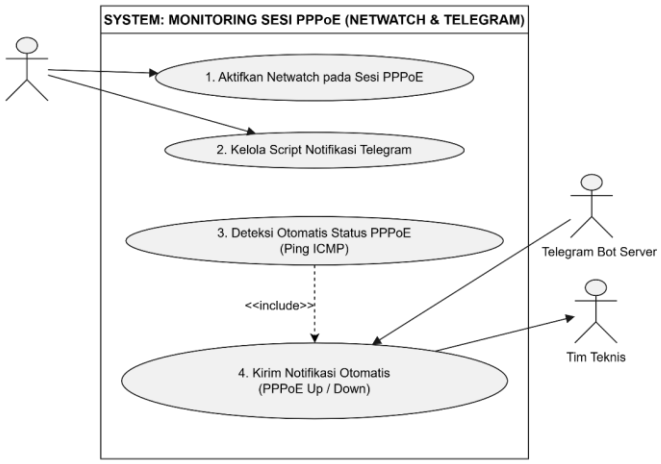
Sistem terdiri atas sesi PPPoE pelanggan, PPP Profile sebagai pemicu *event*, *Netwatch* sebagai pemantau ketersediaan host, dan Telegram Bot API sebagai kanal notifikasi. Setelah autentikasi berhasil, *RouterOS* menyediakan *username* dan *Remote IP* sesi. *Event On Up* menggunakan informasi tersebut untuk mengirim notifikasi dan menambahkan target Netwatch dengan interval 10 detik serta timeout 1000 milidetik. *Netwatch* kemudian mengirimkan ICMP Echo Request secara periodik. Ketika sesi berakhir atau target tidak merespons, event On Down mengirim notifikasi pemutusan dan menghapus entri *Netwatch* berdasarkan identitas akun.



Gambar 2. Arsitektur konseptual monitoring sesi PPPoE dinamis

Gambar 2 memperlihatkan hubungan logis antara pelanggan aktif, *Core Router MikroTik*, skrip internal, *Netwatch*, Telegram *Bot Cloud*, dan tim teknis. Remote IP, waktu, dan status sesi menjadi informasi yang diproses oleh skrip; perubahan status kemudian memicu notifikasi dan tindakan perbaikan.

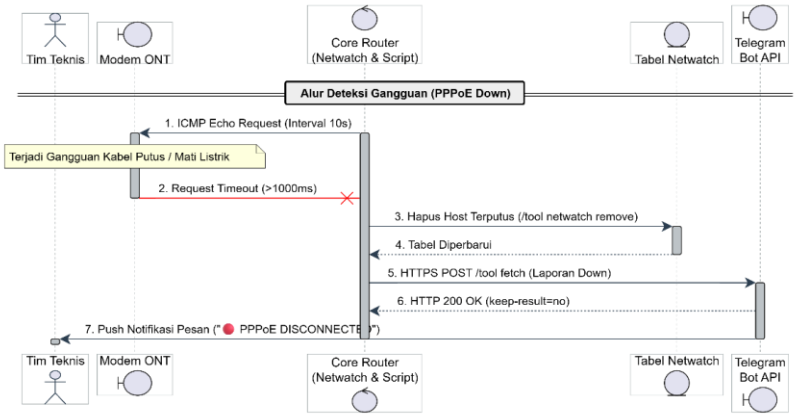
Interaksi fungsional sistem dimodelkan melalui aktor administrator sistem, tim teknis, dan layanan Telegram Bot. Administrator mengaktifkan *Netwatch* serta mengelola skrip, sedangkan sistem mendeteksi status PPPoE dan mengirimkan notifikasi otomatis kepada tim teknis.



Gambar 3. Use case diagram sistem monitoring sesi PPPoE

Gambar 3 memetakan empat fungsi utama sistem, yaitu aktivasi *Netwatch* pada sesi PPPoE, pengelolaan skrip Telegram, deteksi status melalui ping ICMP, dan pengiriman notifikasi PPPoE *Up/Down*. Aktor administrator mengelola konfigurasi, sedangkan tim teknis menerima keluaran monitoring.

Urutan proses gangguan dimodelkan secara kronologis mulai dari terjadinya gangguan pada kabel atau perangkat pelanggan, perubahan status sesi pada router, eksekusi event handler, hingga penerimaan notifikasi oleh Telegram Bot dan tim teknis.



Gambar 4. *Sequence* diagram deteksi gangguan PPPoE

Gambar 4 menunjukkan urutan pertukaran kejadian ketika kabel terputus atau perangkat pelanggan kehilangan daya. *Event* pada router diproses oleh *event handler*, kemudian pesan gangguan diteruskan melalui Telegram Bot kepada tim teknis.

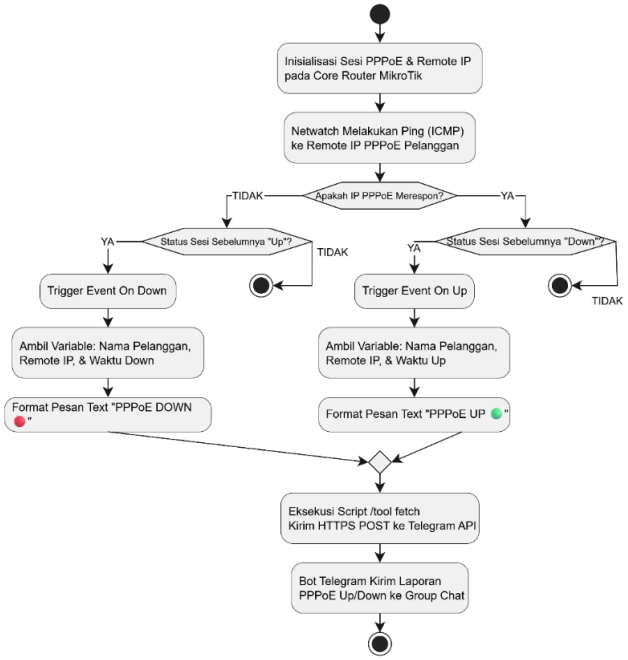
Tabel 2. Arsitektur fungsional sistem monitoring

Komponen	Masukan	Proses	Keluaran
Sesi PPPoE	Kredensial pelanggan	Autentikasi dan pembentukan sesi	Username dan Remote IP
PPP Profile: On Up	Username dan Remote IP	Bentuk pesan, kirim notifikasi, tambah host	Pesan CONNECTED dan target Netwatch

Komponen	Masukan	Proses	Keluaran
Netwatch	Target Remote IP	ICMP Echo Request setiap 10 detik	Status host Up atau Down
PPP Profile: On Down	Event sesi berakhir	Bentuk pesan dan hapus host terkait	Pesan DISCONNECTED dan daftar bersih
Telegram Bot API	Permintaan HTTPS dari router	Meneruskan pesan ke grup teknis	Notifikasi pada perangkat seluler

Tabel 2 menjelaskan pemetaan masukan, proses, dan keluaran pada setiap komponen sistem. Pemetaan tersebut memperlihatkan bahwa Remote IP menjadi penghubung antara sesi PPPoE dan target *Netwatch*, sementara Telegram Bot API menjadi kanal keluaran bagi tim teknis.

2.4 Prosedur implementasi



Gambar 5. Flowchart sistem monitoring sesi PPPoE

Gambar 5 menunjukkan keputusan sistem ketika hasil ping ICMP diterima atau tidak diterima. Status sebelumnya digunakan untuk membedakan pemicu *On Up dan On Down*; setelah data sesi diformat, */tool fetch* mengirimkan pesan ke Telegram Bot API.

Tabel 3. Tahapan implementasi sistem

Tahap	Aktivitas	Luaran
1	Menganalisis kebutuhan monitoring PPPoE dan notifikasi	Kebutuhan fungsional
2	Merancang alur On Up, On Down, Netwatch, dan Telegram	Rancangan arsitektur
3	Menempatkan event handler pada PPP Profile	Skrip aktif pada siklus sesi
4	Mengatur interval 10 detik dan timeout 1000 ms	Target Remote IP terpantau
5	Mengintegrasikan Telegram Bot API melalui HTTPS	Notifikasi status koneksi
6	Menjalankan skenario koneksi dan pemutusan	Data pengujian fungsional
7	Mengamati ICMP menggunakan Wireshark	Bukti pertukaran paket

Tabel 3 menunjukkan tahapan implementasi dari analisis kebutuhan hingga inspeksi paket. Urutan tersebut digunakan untuk menjaga keterlacakan antara kebutuhan fungsional, konfigurasi event handler, pengujian status sesi, dan bukti komunikasi jaringan.

2.5 Skenario dan parameter pengujian

Pengujian fungsional dilakukan melalui tiga kondisi operasional yang diturunkan menjadi empat item uji: pemutusan koneksi, pemulihan koneksi, penambahan target *Netwatch*, dan penghapusan target *Netwatch*. Pengamatan Wireshark menggunakan lalu lintas ICMP pada alamat uji yang dianonimkan. Parameter kinerja mengacu pada hasil pengujian yang dicatat dalam dokumen penelitian. Informasi kredensial, token, chat ID, dan alamat jaringan operasional tidak disertakan dalam artikel.

Tabel 4. Skenario pengujian fungsional

Kode	Kondisi uji	Indikator keberhasilan
I	Modem mati atau sesi PPPoE terputus	Pesan DISCONNECTED diterima dengan identitas akun, IP, dan waktu
II	Modem menyala dan sesi terbentuk kembali	Pesan CONNECTED diterima dengan rincian sesi terbaru
III-A	Sesi PPPoE baru terbentuk	Remote IP muncul pada Netwatch dengan interval 10 detik
III-B	Sesi PPPoE terputus	Entri terkait dihapus tanpa menghapus target lain

Tabel 4 menetapkan kondisi masukan dan indikator keberhasilan setiap skenario. Empat item uji mencakup notifikasi pemutusan, notifikasi pemulihan, penambahan target, dan penghapusan target tanpa intervensi manual pada tabel *Netwatch*.

2.6 Teknik pengumpulan dan analisis data

Pengumpulan data dilakukan melalui observasi terstruktur pada siklus sesi PPPoE. Tahap pertama adalah membentuk sesi pelanggan dan mencatat username, Remote IP, waktu *event On Up*, serta kemunculan target pada *Netwatch*. Tahap kedua adalah mensimulasikan gangguan dengan memutus kabel atau mematikan perangkat pelanggan, kemudian mencatat waktu *event On Down*, status target, dan pesan yang diterima melalui Telegram. Tahap ketiga adalah memulihkan sesi untuk memeriksa notifikasi pemulihan dan penambahan ulang target.

Validasi komunikasi jaringan dilakukan dengan menangkap paket pada segmen pengujian menggunakan *Wireshark*. Filter ICMP digunakan untuk membandingkan *Echo Request* dan *Echo Reply* pada kondisi normal serta ketiadaan respons pada kondisi gangguan. Permintaan HTTPS ke Telegram diamati hanya dari sisi waktu dan status penyampaian; token autentikasi serta alamat operasional tidak dimasukkan ke dalam naskah.

Data fungsional dianalisis secara deskriptif berdasarkan kesesuaian keluaran terhadap indikator pada Tabel 4. Data QoS dihitung menggunakan jumlah paket, waktu antar-paket, dan waktu deteksi-penerimaan yang tersedia pada catatan pengujian. Apabila durasi transmisi tidak tersedia, nilai throughput tidak dipaksakan; penelitian hanya melaporkan ukuran payload dan menjelaskan keterbatasan pengukuran tersebut.

3. Hasil dan Pembahasan

3.1 Implementasi event handler PPP Profile

Event handler ditempatkan pada PPP Profile yang digunakan oleh sesi PPPoE pelanggan. Pada *event On Up*, variabel *username* dan *Remote IP* digunakan untuk membentuk pesan berstatus *CONNECTED*. Pesan dikirimkan melalui */tool fetch* ke Telegram Bot API. Setelah itu, Remote IP didaftarkan sebagai target Netwatch dengan interval 10 detik dan timeout 1000 milidetik. Cara ini menghilangkan kebutuhan memasukkan alamat IP pelanggan secara manual.

Pada *event On Down*, sistem membentuk pesan berstatus *DISCONNECTED* yang memuat identitas akun, Remote IP, dan waktu kejadian. Sistem kemudian mencari entri Netwatch berdasarkan comment akun dan menghapus entri yang terkait. Mekanisme ini menjaga daftar target agar mengikuti sesi yang sedang aktif. Pada implementasi produksi, identitas berbasis username perlu diperkuat dengan kombinasi username dan Remote IP apabila satu akun dapat memiliki beberapa sesi bersamaan.

3.2 Hasil pengujian fungsional

Pengujian *black-box* menunjukkan bahwa empat item uji memperoleh status berhasil. *Event Down* menghasilkan notifikasi pemutusan, *event Up* menghasilkan notifikasi pemulihan, *Remote IP* ditambahkan ketika sesi terbentuk, dan entri terkait dihapus ketika sesi berakhir.

Tabel 5. Hasil pengujian fungsional sistem

Skenario	Kondisi	Hasil pengujian	Status
I	Deteksi gangguan (Down)	Pesan DISCONNECTED diterima dengan data akun, IP, dan waktu putus	Berhasil
II	Deteksi pemulihan (Up)	Pesan CONNECTED diterima dengan rincian sesi terbaru	Berhasil
III-A	Penambahan host Netwatch	Host baru muncul dengan Remote IP, interval 10 detik, dan comment akun	Berhasil
III-B	Penghapusan host Netwatch	Host terkait terhapus tanpa mengganggu target lain	Berhasil

Tabel 5 memperlihatkan bahwa seluruh skenario fungsional memperoleh status berhasil. Dua skenario pertama memvalidasi jalur notifikasi, sedangkan dua skenario terakhir memvalidasi konsistensi pengelolaan target *Netwatch* selama perubahan sesi PPPoE.

3.3 Analisis paket ICMP

Analisis *Wireshark* memperlihatkan ICMP *Echo Request* yang dikirim dari router menuju Remote IP pelanggan. Pada sesi capture yang dilaporkan, panjang paket berada pada kisaran 98–102 *byte* dengan TTL 63–64. Ketika koneksi terganggu, *Echo Request* tidak memperoleh *Echo Reply* dan *Wireshark* menandai paket sebagai *no response found.* Kondisi tersebut menjadi indikator hilangnya keterjangkauan host dan memicu perubahan status *Netwatch* setelah ambang timeout terpenuhi.

Hasil ini mendukung hubungan antara kegagalan respons ICMP dan eksekusi *event On Down* pada skenario yang diuji. Interpretasi tetap dibatasi pada *capture* yang tersedia karena *firewall*, *rate limiting* ICMP, atau kebijakan perangkat antara dapat menyebabkan host tidak merespons ICMP meskipun layanan tertentu masih berjalan.

Tabel 6. Hasil observasi paket ICMP

Kondisi	Paket yang diamati	Interpretasi
Koneksi tersedia	Echo Request diikuti Echo Reply	Host terjangkau dan tetap berstatus Up
Koneksi terganggu	Echo Request tanpa Echo Reply	Status host dapat berubah menjadi Down setelah timeout

Tabel 6 membandingkan kondisi ketika *Echo Reply* diterima dan ketika respons ICMP tidak ditemukan. Perbedaan tersebut menjadi dasar interpretasi status keterjangkauan *host* pada *Netwatch*, dengan catatan bahwa pemblokiran ICMP dapat memengaruhi hasil observasi.

3.4 Evaluasi kinerja dan notifikasi

Evaluasi kinerja menunjukkan *delay* ICMP sebesar 3,18 ms, *packet loss* 0% pada kondisi operasi normal, *jitter* 1,12 ms, dan ukuran payload notifikasi sekitar 1,35 KB per pesan. Pada simulasi koneksi terputus, *packet loss* tercatat 100% sampai status *host* melewati ambang *timeout*. Latensi pengiriman notifikasi dari deteksi router sampai pesan diterima pada Telegram berada pada rentang 1,40–1,80 detik. Nilai tersebut berada di bawah target 2 detik yang ditetapkan pada rancangan penelitian.

Perhitungan dilakukan berdasarkan definisi operasional setiap parameter. Nilai delay dan jitter merupakan nilai rata-rata yang dilaporkan pada hasil pengujian, sedangkan packet loss dihitung dari perbandingan jumlah paket dikirim dan diterima.

$$\textit{Delay rata-rata} = \Sigma\, RTT_i / N = 3{,}18 \text{ ms} \quad (1)$$

$$\textit{Packet Loss (\%)} = (\text{(Paket dikirim} - \text{Paket diterima)} / \text{Paket dikirim}) \times 100\% \quad (2)$$

Pada kondisi normal, substitusinya adalah ((N − N) / N) × 100% = 0%. Pada simulasi link down, tidak ada Echo Reply sehingga substitusinya adalah ((N − 0) / N) × 100% = 100%.

$$\textit{Jitter} = \Sigma\, |RTT_i - RTT_{i-1}| / (N - 1) = 1{,}12 \text{ ms} \quad (3)$$

$$\textit{Latensi notifikasi} = t_\text{diterima} - t_\text{terdeteksi} = 1{,}40\text{–}1{,}80 \text{ s} \quad (4)$$

$$\textit{Throughput} = \text{jumlah data diterima } (bit) / \text{waktu transmisi } (s) \quad (5)$$

$$\textit{Payload} = 1{,}35 \text{ KB} \times 1.024 \text{ byte/KB} \times 8 \text{ bit/byte} = 11.059{,}2 \text{ bit} \approx 11{,}06 \text{ kbit/pesan} \quad (6)$$

Data penelitian mencatat ukuran *payload* sekitar 1,35 KB per notifikasi, bukan durasi transmisi. Karena waktu transmisi tidak tersedia, laju throughput dalam bit per detik tidak dapat dihitung secara sahih. Nilai 11,06 kbit merupakan ukuran data per pesan dan tidak boleh dilabeli sebagai *throughput rate*.

Tabel 7. Hasil evaluasi kinerja sistem

Parameter	Skenario	Hasil	Interpretasi
Delay/latency	ICMP Netwatch	$\Sigma\ RTT_i / N = 3{,}18$ ms	<150 ms; sangat baik
Packet loss	Kondisi normal	$((N-N)/N) \times 100\% = 0\%$	Tidak ada kehilangan paket
Packet loss	Simulasi link down	$((N-0)/N) \times 100\% = 100\%$	Tidak ada Echo Reply
Jitter	Variasi ICMP	$\Sigma\|RTT_i - RTT_{i-1}\|/(N-1) = 1{,}12$ ms	<75 ms; fluktuasi rendah
Payload notifikasi	Permintaan HTTPS	1,35 KB = 11,06 kbit/pesan	Ukuran payload, bukan laju throughput
Throughput rate	Permintaan HTTPS	Tidak dihitung; durasi transmisi tidak tersedia	Rumus tersedia pada Pers. (3.5)
Latensi notifikasi	Router ke Telegram	t_diterima−t_terdeteksi = 1,40–1,80 s	<2 s; sesuai target

Tabel 7 merangkum hasil substitusi rumus QoS dan membedakan metrik yang benar-benar terukur dari ukuran *payload* pesan. *Delay, packet loss,* dan *jitter* dihitung dari pertukaran ICMP, sedangkan latensi notifikasi dihitung dari selisih waktu deteksi dan waktu penerimaan pesan.

Hasil tersebut menunjukkan bahwa integrasi *Netwatch* dan Telegram dapat menyediakan peringatan dini pada skenario pengujian. Namun, hasil tidak boleh ditafsirkan sebagai ukuran reliabilitas jangka panjang karena dokumen penelitian belum mencantumkan jumlah pengulangan, jumlah akun uji, distribusi kegagalan, atau interval kepercayaan. Pengujian lanjutan perlu mengukur rasio notifikasi berhasil, waktu deteksi secara terpisah dari waktu pengiriman, perilaku *reconnect*, *multi-session,* serta beban CPU dan memori router.

3.5 Keterbatasan dan implikasi

Kontribusi utama sistem adalah sinkronisasi otomatis antara sesi PPPoE dan target *Netwatch*. Telegram Bot API menyediakan kanal notifikasi yang dapat diterima secara mobile tanpa aplikasi monitoring khusus. Meskipun demikian, keberhasilan pengiriman bergantung pada konektivitas router ke Telegram, resolusi DNS, validitas token, dan ketersediaan layanan eksternal. Identitas entri berdasarkan *comment username* juga perlu ditinjau kembali apabila jaringan mengizinkan beberapa sesi dengan akun yang sama.

Penelitian ini tidak menyertakan *token bot, chat* ID, alamat IP operasional, atau konfigurasi rahasia. Pengelolaan kredensial harus dilakukan melalui penyimpanan rahasia dan rotasi token apabila kredensial pernah tercantum dalam dokumen kerja. Pengembangan berikutnya dapat menambahkan *dashboard* riwayat downtime, notifikasi dua arah, dan integrasi helpdesk setelah mekanisme dasar diuji pada skala sesi yang lebih besar.

3.6 Implikasi operasional dan validitas hasil

Dari sudut pandang operasional, integrasi *event handler* dan *Netwatch* membentuk satu rantai pengawasan yang mengikuti perubahan sesi. *Event On Up* tidak hanya mengirimkan pesan, tetapi juga menyiapkan target pemantauan. Sebaliknya, *event On Down* menghapus target yang tidak lagi relevan setelah menyampaikan informasi gangguan. Pola ini mengurangi risiko daftar monitoring berisi alamat lama dan membantu tim teknis memusatkan perhatian pada sesi yang masih aktif.

Keandalan keluaran tetap dipengaruhi oleh beberapa dependensi eksternal. *Netwatch* memerlukan respons ICMP, sedangkan notifikasi memerlukan akses HTTPS dari router ke Telegram Bot API dan konektivitas perangkat penerima. Oleh sebab itu, status Down harus dipahami sebagai tidak diterimanya respons sesuai interval dan timeout yang dikonfigurasi, bukan sebagai bukti tunggal bahwa media fisik pasti terputus. Korelasi dengan log PPPoE dan pemeriksaan lapangan tetap diperlukan untuk diagnosis final.

Validitas hasil pada penelitian ini bersifat kontekstual karena pengujian dilakukan pada satu lingkungan *Core Router* dan catatan sumber tidak menyediakan jumlah repetisi statistik untuk setiap skenario. Hasil yang dilaporkan cukup untuk memverifikasi alur fungsional dan karakteristik paket yang diamati, tetapi belum dapat digeneralisasi sebagai angka kinerja seluruh jaringan. Pengujian lanjutan sebaiknya menambah jumlah repetisi, variasi beban pelanggan, gangguan selektif, dan pencatatan timestamp otomatis agar interval kepercayaan serta throughput aktual dapat dihitung.

4. Kesimpulan

Sistem monitoring PPPoE berbasis Telegram Bot API dan *Netwatch* berhasil diimplementasikan pada *Core Router* MikroTik Erlangga Network. *Event On Up* mengambil *username* dan *Remote IP*, mengirimkan notifikasi, serta mendaftarkan target secara otomatis. *Event On Down* mengirimkan notifikasi pemutusan dan menghapus entri pemantauan terkait.

Pengujian fungsional berhasil pada skenario deteksi *Down*, pemulihan *Up*, penambahan host *Netwatch*, dan penghapusan host Netwatch. Analisis paket mendukung mekanisme deteksi melalui ICMP *Echo Request* dan ketiadaan *Echo Reply* pada kondisi gangguan.

Pada skenario pengujian, *delay* tercatat 3,18 ms, *jitter* 1,12 ms, *packet loss* 0% pada kondisi normal, dan latensi notifikasi 1,40–1,80 detik. Hasil ini menunjukkan bahwa sistem mampu menyediakan peringatan dini dengan *respons* di bawah dua detik pada lingkungan yang diuji. Validasi jangka panjang dan pengujian *multi-session* masih diperlukan sebelum hasil digeneralisasi ke seluruh pelanggan.

Ucapan Terima Kasih

Penulis menyampaikan terima kasih kepada Erlangga Network (PT. Baraya Bareng Sadulur) atas izin dan dukungan dalam pelaksanaan penelitian pada lingkungan jaringan.

Pernyataan Etik dan Deklarasi Artikel

Etika penelitian

Penelitian berfokus pada observasi infrastruktur jaringan dan pengujian perangkat. Naskah tidak menyertakan identitas pelanggan, isi kredensial, token bot, atau alamat operasional yang bersifat sensitif.

Kontribusi penulis

Mela Puspita Sari berkontribusi pada perancangan, implementasi, pengujian, analisis, dan penyusunan naskah penelitian.

Penggunaan kecerdasan buatan

Alat bantu kecerdasan buatan digunakan secara terbatas untuk membantu penyuntingan bahasa dan penataan naskah. Seluruh interpretasi teknis, pemilihan data, dan tanggung jawab ilmiah tetap berada pada penulis.

Pendanaan dan konflik kepentingan

Informasi pendanaan dan deklarasi konflik kepentingan tidak tercantum dalam dokumen sumber. Kedua informasi tersebut perlu dikonfirmasi oleh penulis sebelum naskah dikirim ke jurnal.

Daftar Pustaka

[1] Jufri and Heryanto, “Pemanfaatan Fitur Netwatch MikroTik dalam Monitoring Ketersediaan Jaringan Local Area Network,” Jurnal Teknik Informatika dan Jaringan, 2021.
[2] Adi and Athallah, “Implementasi Fitur Scripting dan Otomatisasi Jaringan pada MikroTik RouterOS untuk Efisiensi Penanganan Gangguan,” Jurnal Rekayasa Teknologi dan Sistem Komputer, 2024.
[3] A. Pratomo, “Pengembangan Sistem Pengawasan dan Otomatisasi Jaringan Komputer Berbasis MikroTik RouterOS,” Jurnal Ilmu Komputer dan Jaringan, 2023.
[4] Samad and Persadha, “Memahami Perang Siber Rusia dan Peran Badan Intelijen Negara dalam Menangkal Ancaman Siber,” Jurnal IPTEK-KOM, pp. 135–146, 2022.
[5] T. Hidayat and R. Nugraha, “Perancangan Infrastruktur Jaringan FTTH Menggunakan Teknologi GPON untuk Layanan Triple Play,” Jurnal Ilmiah Teknik Elektro Komputer dan Informatika, pp. 205–217, 2022.
[6] E. Rahmawati and Z. Arifin, Komunikasi Data dan Fondasi Infrastruktur Jaringan. Yogyakarta: Deepublish, 2022.
[7] M. Riduan and R. Sadikin, “Analisis Downtime dan Manajemen Risiko Infrastruktur Jaringan pada Perusahaan Penyedia Layanan Internet,” Jurnal Teknologi Informasi dan Komunikasi, 2021.
[8] D. Kurniawan and A. Syahputra, “Evaluasi Keamanan dan Kinerja Komunikasi Data Berbasis Enkapsulasi Protokol TCP/IP,” Jurnal Riset Sistem Informasi dan Teknik Informatika, pp. 101–112, 2023.
[9] D. Nurcahyani, S. Wibowo, and H. Prasetyo, “Analisis Keamanan Pertukaran Data Berbasis Protokol Jaringan Komputer Menggunakan Wireshark Packet Analyzer,” Jurnal Ilmiah Komputasi dan Teknologi Informasi, pp. 112–123, 2023.

[10] A. Pratama and D. Setiawan, "Penerapan Unified Modeling Language dalam Perancangan Sistem Pengawasan Infrastruktur Jaringan Berbasis Web," Jurnal Sistem Informasi dan Rekayasa Komputer, pp. 67–78, 2022.
[11] R. Putra and S. Darmawan, "Komparasi Algoritma Autentikasi PAP dan CHAP terhadap Keamanan Akses Jaringan Komputer," Jurnal Sains Komputer dan Informatika, pp. 654–663, 2022.
[12] D. Setiawan and M. Fauzi, "Analisis Parameter Quality of Service (QoS) pada Pengujian Latensi Jaringan Menggunakan Protokol ICMP," Jurnal Rekayasa Informasi, pp. 55–63, 2021.
[13] I. Wahyudi and A. Firmansyah, "Rancang Bangun Server PPPoE Terpusat Berbasis Manajemen RADIUS untuk Otentikasi Pelanggan ISP," Jurnal Teknologi Terpadu, pp. 110–120, 2022.
[14] R. Wulandari and A. Pratama, "Evaluasi Kinerja Jaringan Komputer Menggunakan Metode Quality of Service Standar TIPHON pada Layanan Akses Internet," Jurnal Ilmiah Komputasi dan Jaringan Terdistribusi, pp. 104–115, 2022.
[15] A. S. Rosa and M. Shalahuddin, Rekayasa Perangkat Lunak Terstruktur dan Berorientasi Objek. Bandung: Informatika, 2021.