Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

Integrasi Zabbix dan Network Automation untuk Pengelolaan Bandwidth MikroTik

I Made Diva Dwipayana, Gede Arna Jude Saskara*, P Wayan Arta Suyasa

Fakultas Teknik dan Kejuruan, Prodi Pendidikan Teknik Informatika, Universitas Pendidikan Ganesha, Singaraja, Indonesia
Email: [1]diva.dwipayana@student.undiksha.ac.id, [2,*]jude.saskara@undiksha.ac.id, [3]arta.suyasa@undiksha.ac.id
Email Penulis Korespondensi: jude.saskara@undiksha.ac.id

Abstrak−Milk & Madu Group memiliki infrastruktur jaringan yang tersebar pada 12 cabang dengan sekitar 63 perangkat jaringan. Pengelolaan dan pemantauan bandwidth yang masih dilakukan secara manual menyebabkan proses identifikasi gangguan dan konfigurasi jaringan membutuhkan waktu relatif lama, terutama ketika terjadi peningkatan trafik pada beberapa cabang secara bersamaan. Penelitian ini bertujuan mengimplementasikan sistem monitoring jaringan berbasis Zabbix yang terintegrasi dengan network automation menggunakan Python untuk mendukung otomasi pengelolaan bandwidth pada perangkat MikroTik. Penelitian menggunakan metode Research and Development (R&D) dengan model Prototype. Sistem dikembangkan dengan mengintegrasikan Zabbix Server, backend Flask, Telegram Bot, database SQLite, serta library Paramiko melalui koneksi SSH. Mekanisme otomasi memanfaatkan trigger dan webhook pada Zabbix untuk mendeteksi kondisi jaringan, kemudian meneruskan informasi tersebut ke backend untuk diproses sebelum konfigurasi bandwidth dijalankan pada perangkat MikroTik dengan persetujuan administrator melalui Telegram Bot. Hasil pengujian fungsional menunjukkan seluruh fitur sistem berjalan sesuai kebutuhan dan seluruh komponen dapat terintegrasi dengan baik. Pengujian User Acceptance Test (UAT) memperoleh tingkat penerimaan pengguna sebesar 96%. Pengujian performa menunjukkan efisiensi waktu konfigurasi sebesar 54,59% pada workflow problem dan 76,56% pada workflow recovery, dengan tingkat keberhasilan masing-masing sebesar 96,67% dan 98,33%. Hasil penelitian menunjukkan bahwa integrasi monitoring dan network automation dapat meningkatkan efisiensi serta keandalan pengelolaan jaringan multi-cabang, sekaligus membantu administrator melakukan konfigurasi secara lebih cepat dan terkontrol.

Kata Kunci: Zabbix; Network Automation; Monitoring Jaringan; Manajemen Bandwidth; MikroTik

Abstract−Milk & Madu Group operates a network infrastructure distributed across 12 branches with approximately 63 network devices. Manual bandwidth management and monitoring make the identification of network problems and configuration processes relatively time-consuming, particularly when traffic increases simultaneously across multiple branches. This study aims to implement a Zabbix-based network monitoring system integrated with network automation using Python to support automated bandwidth management on MikroTik devices. The research employed a Research and Development (R&D) method with a Prototype model. The system was developed by integrating Zabbix Server, a Flask backend, Telegram Bot, SQLite database, and the Paramiko library through SSH connections. The automation mechanism utilizes Zabbix triggers and webhooks to detect network conditions and forward the information to the backend for processing before bandwidth configuration is executed on MikroTik devices with administrator approval through Telegram Bot. Functional testing results indicate that all system features operate as required and all components are well integrated. User Acceptance Test (UAT) achieved a user acceptance rate of 96%. Performance testing showed configuration time efficiency of 54.59% in the problem workflow and 76.56% in the recovery workflow, with success rates of 96.67% and 98.33%, respectively. The results demonstrate that integrating network monitoring and network automation can improve the efficiency and reliability of multi-branch network management while enabling administrators to perform configurations more quickly and in a controlled manner.

Keywords: Zabbix; Network Automation; Network Monitoring; Bandwidth Management; MikroTik

1. PENDAHULUAN

Pertumbuhan perangkat jaringan dan tingginya ketergantungan pada layanan digital menuntut organisasi untuk menerapkan pengelolaan infrastruktur yang lebih efektif dan terpusat [1]. Performa jaringan yang terjaga dengan baik merupakan kunci utama dalam menjamin kelancaran operasional IT, sebab penurunan kualitas jaringan berdampak langsung pada pengalaman pengguna [2]. Selain itu, ketersediaan jaringan yang tinggi juga diperlukan untuk menjamin keberlangsungan layanan dan akses informasi pada lingkungan organisasi modern [3]. Oleh karena itu, diperlukan mekanisme pengelolaan jaringan yang mampu memastikan performa layanan tetap optimal serta mendukung proses identifikasi gangguan secara cepat dan akurat [4].

Pemantauan jaringan dilakukan secara berkelanjutan untuk mengamati kondisi perangkat dan layanan, sekaligus memastikan performa serta ketersediaannya tetap terjaga [5]. Data yang diperoleh dari proses ini menjadi dasar dalam menganalisis masalah serta menentukan tindakan korektif yang tepat [4]. Untuk mempermudah analisis tersebut, sistem pemantauan umumnya menyajikan informasi secara visual melalui *dashboard* guna mendukung pengambilan keputusan oleh administrator [6]. Zabbix menjadi salah satu platform yang banyak dipilih karena mengusung pemantauan *real-time*, sistem *trigger*, notifikasi, serta fleksibilitas integrasi dengan layanan eksternal [7]. Zabbix juga telah terbukti efektif dalam mendukung proses monitoring infrastruktur jaringan pada lingkungan yang kompleks dan berskala besar [7]. Pemanfaatan sistem monitoring yang baik dapat membantu administrator mengidentifikasi penurunan kualitas layanan sebelum berdampak pada pengguna akhir [8]. Pemanfaatan sistem monitoring yang terintegrasi dengan mekanisme notifikasi memungkinkan administrator memperoleh informasi gangguan jaringan secara lebih cepat sehingga proses penanganan dapat dilakukan secara lebih responsif [9].

Milk & Madu Group memiliki 12 cabang dengan sekitar 63 perangkat jaringan yang dikelola oleh dua staf teknologi informasi. Berdasarkan hasil observasi, proses monitoring jaringan masih dilakukan secara manual

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

menggunakan utilitas bawaan MikroTik pada masing-masing perangkat. Proses pengelolaan bandwidth juga masih memerlukan konfigurasi langsung oleh administrator ketika terjadi perubahan kondisi jaringan. Kondisi tersebut menyebabkan pengelolaan jaringan sangat bergantung pada ketersediaan staf IT, sehingga proses pemantauan, identifikasi gangguan, dan penyesuaian konfigurasi memerlukan waktu yang relatif lebih lama. Selain berdampak pada efisiensi operasional, tingginya keterlibatan administrator dalam proses konfigurasi juga berpotensi meningkatkan risiko terjadinya kesalahan konfigurasi (*misconfiguration*). Penelitian Listartha dan Saskara [10] menunjukkan bahwa *misconfiguration* merupakan salah satu kerentanan pada perangkat jaringan yang dapat dieksploitasi untuk mengganggu ketersediaan dan stabilitas layanan jaringan. Oleh karena itu, diperlukan solusi yang mampu membantu administrator dalam melakukan pemantauan dan konfigurasi jaringan secara lebih cepat, konsisten, dan terkontrol.

Pengelolaan bandwidth merupakan salah satu aspek penting dalam menjaga kualitas layanan jaringan karena berkaitan langsung dengan distribusi sumber daya jaringan kepada pengguna [11]. Manajemen bandwidth yang tidak optimal dapat menyebabkan ketimpangan penggunaan bandwidth dan menurunkan kualitas akses jaringan pada pengguna lain [12]. Salah satu metode yang banyak digunakan pada perangkat MikroTik adalah *Per Connection Queue* (PCQ) karena mampu membagi bandwidth secara lebih merata kepada setiap koneksi aktif [13]. Penelitian Auriga et al. [12] menunjukkan bahwa metode PCQ memiliki performa yang baik dalam mendukung pengelolaan bandwidth pada jaringan berbasis MikroTik. Selain itu, penelitian Dwipayoga et al. [14] menunjukkan bahwa otomasi pada manajemen bandwidth dapat membantu administrator dalam mengelola konfigurasi jaringan secara lebih efisien. Sistem automasi merupakan teknologi yang mengintegrasikan sistem komputerisasi dengan program tertentu untuk menjalankan suatu proses secara otomatis dan dilengkapi mekanisme umpan balik guna memastikan instruksi dapat berjalan sesuai tujuan [15]. Kebutuhan akan pengelolaan jaringan yang semakin cepat dan konsisten turut mendorong pemanfaatan *network automation* pada berbagai lingkungan jaringan modern [14]. Dengan adanya otomasi, proses konfigurasi, pemeliharaan, hingga pengelolaan perangkat dapat dilakukan dengan lebih cepat dibandingkan jika seluruh pekerjaan dilakukan secara manual [16]. Mazin et al. [17] menjelaskan bahwa Python dapat digunakan untuk berinteraksi langsung dengan perangkat jaringan dan menjalankan konfigurasi secara otomatis melalui koneksi SSH. Penggunaan SSH sendiri memberikan mekanisme komunikasi yang aman antara sistem dan perangkat jaringan [17]. Penerapan otomasi juga terbukti mampu mengurangi pekerjaan administratif yang berulang sehingga administrator dapat lebih fokus pada aktivitas yang bersifat strategis [18]. Selain meningkatkan efisiensi operasional, *network automation* dapat membantu menekan potensi kesalahan konfigurasi yang sering muncul akibat faktor manusia dalam proses pengelolaan jaringan [14].

Beberapa penelitian terdahulu telah membahas penerapan monitoring jaringan dan otomasi jaringan. Fachrurrozi et al. [19] mengembangkan sistem monitoring jaringan berbasis LibreNMS yang terintegrasi dengan Telegram dan email sebagai media notifikasi. Guo et al. [20] memanfaatkan Zabbix sebagai platform monitoring untuk mendukung pengelolaan jaringan berskala besar. Santyadiputra et al. [21] mengembangkan simulasi *Automatic Network Administration* (ANA) untuk mendukung proses administrasi jaringan secara otomatis. Sementara itu, Dwipayoga et al. [14] mengembangkan sistem network automation untuk manajemen bandwidth MikroTik menggunakan metode *Hierarchical Token Bucket* (HTB). Meskipun penelitian terkait monitoring jaringan dan *network automation* telah banyak dilakukan, keduanya umumnya masih diterapkan sebagai solusi yang berdiri sendiri. Padahal, informasi yang dihasilkan oleh sistem monitoring berpotensi dimanfaatkan sebagai dasar untuk menjalankan tindakan otomatis pada perangkat jaringan. Belum terintegrasinya monitoring jaringan, pengelolaan bandwidth, dan *network automation* dalam satu mekanisme yang saling terhubung menunjukkan adanya peluang pengembangan, terutama pada lingkungan jaringan multi-cabang yang membutuhkan respons cepat terhadap perubahan kondisi jaringan.

Oleh karena itu, penelitian ini mengimplementasikan sistem monitoring jaringan berbasis Zabbix yang terintegrasi dengan *network automation* menggunakan Python untuk mendukung pengelolaan bandwidth pada perangkat MikroTik. Kebaruan penelitian terletak pada pemanfaatan data monitoring sebagai pemicu proses otomasi melalui mekanisme *trigger* dan *webhook*, yang kemudian diproses oleh *backend* Flask sebelum mengeksekusi konfigurasi pada perangkat jaringan melalui koneksi SSH. Selain itu, sistem menerapkan mekanisme persetujuan administrator melalui Telegram Bot untuk menjaga kontrol terhadap perubahan konfigurasi yang dilakukan. Dengan pendekatan tersebut, sistem diharapkan mampu meningkatkan efisiensi pengelolaan jaringan, mempercepat respons terhadap gangguan, dan mendukung pengelolaan jaringan multi-cabang secara lebih efektif.

2. METODOLOGI PENELITIAN

Penelitian ini menerapkan metode *Research and Development* (R&D) untuk mengembangkan sistem monitoring jaringan yang terintegrasi dengan *network automation* pada lingkungan jaringan Milk & Madu Group. Metode R&D digunakan karena penelitian berfokus pada pengembangan sistem yang disesuaikan dengan kebutuhan dan kondisi operasional pengguna [22]. Pengembangan sistem menggunakan pendekatan *Prototype*, yang memungkinkan proses pengembangan dilakukan secara bertahap berdasarkan kebutuhan pengguna hingga diperoleh sistem yang sesuai dengan kebutuhan operasional [23].

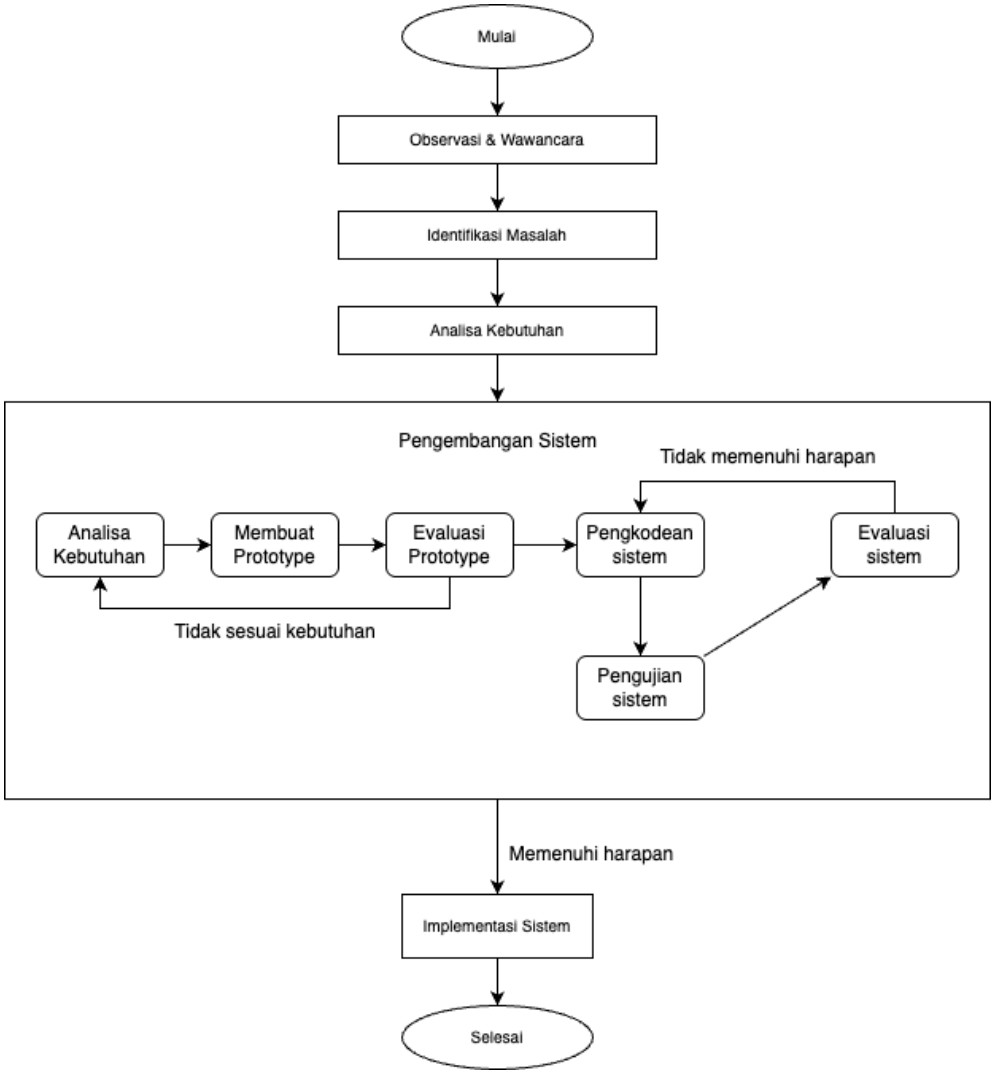
Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

2.1 Objek Penelitian

Objek penelitian adalah infrastruktur jaringan Milk & Madu Group yang terdiri atas 12 cabang dengan sekitar 63 perangkat jaringan. Perangkat yang menjadi fokus penelitian adalah router MikroTik yang digunakan untuk mengelola jaringan dan bandwidth pada masing-masing cabang. Penelitian melibatkan dua staf Teknologi Informasi Milk & Madu Group sebagai pengguna sistem dan responden dalam proses evaluasi sistem. Data yang digunakan dalam pengembangan diperoleh melalui observasi, wawancara, dan dokumentasi untuk mengidentifikasi kondisi jaringan, permasalahan monitoring dan pengelolaan bandwidth, serta kebutuhan pengguna terhadap sistem yang dikembangkan.

2.2 Tahapan Penelitian

Penelitian dilakukan melalui empat tahapan utama, yaitu analisis kebutuhan, perancangan sistem, implementasi sistem, dan pengujian sistem. Tahapan penelitian dirancang untuk menghubungkan permasalahan pengelolaan jaringan dengan penerapan sistem monitoring dan *network automation*. Alur tahapan penelitian beserta penerapan solusi yang dikembangkan ditunjukkan pada Gambar 1.



Gambar 1. Tahapan Penelitian

a. Analisis Kebutuhan
Tahap analisis kebutuhan dilakukan untuk mengidentifikasi kondisi infrastruktur jaringan, proses monitoring, pengelolaan bandwidth, serta permasalahan yang dihadapi administrator pada Milk & Madu Group. Hasil analisis digunakan sebagai dasar dalam menentukan kebutuhan sistem monitoring dan otomasi pengelolaan bandwidth. Tahap ini menghasilkan kebutuhan untuk mengintegrasikan proses monitoring jaringan dengan *network automation*.

b. Perancangan Sistem
Tahap perancangan dilakukan dengan menyusun arsitektur sistem yang mengintegrasikan Zabbix Server, *backend* Flask, Telegram Bot, SQLite, Python, dan perangkat MikroTik. Pada tahap ini dirancang mekanisme *trigger* dan

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

webhook pada Zabbix untuk meneruskan informasi kondisi jaringan ke sistem otomasi. Rancangan tersebut menjadi dasar dalam membangun mekanisme monitoring dan otomasi konfigurasi bandwidth [16].

c. Implementasi Sistem

Tahap implementasi dilakukan dengan mengintegrasikan seluruh komponen berdasarkan rancangan yang telah dibuat. Zabbix digunakan untuk melakukan monitoring jaringan secara *real-time* dan mendeteksi kondisi tertentu melalui mekanisme *trigger*. Informasi dari *trigger* diteruskan melalui *webhook* ke *backend* Flask untuk diproses. Hasil pemrosesan kemudian dikirimkan kepada administrator melalui Telegram Bot untuk memperoleh persetujuan. Setelah persetujuan diberikan, Python menjalankan konfigurasi bandwidth pada perangkat MikroTik melalui koneksi SSH menggunakan library Paramiko [17], [21]. Mekanisme tersebut menghubungkan proses monitoring, pengambilan keputusan, dan konfigurasi jaringan dalam satu alur kerja.

d. Pengujian Sistem

Tahap pengujian dilakukan untuk mengevaluasi fungsi, penerimaan pengguna, dan performa sistem yang telah dikembangkan. Pengujian terdiri atas *Black Box Testing*, *User Acceptance Test* (UAT), dan *Performance Testing*. *Black Box Testing* digunakan untuk memastikan fungsi sistem berjalan sesuai kebutuhan. UAT dilakukan dengan melibatkan dua staf Teknologi Informasi untuk mengetahui tingkat penerimaan pengguna terhadap sistem. *Performance Testing* dilakukan dengan membandingkan waktu konfigurasi bandwidth secara manual dan menggunakan sistem pada kondisi *workflow problem* dan *workflow recovery*. Nilai UAT dihitung berdasarkan perbandingan jumlah skor respons yang diperoleh dengan skor maksimum menggunakan Persamaan (1).

$$\text{User Acceptance } Test = \left(\frac{\sum R}{N}\right) \times 100\% \qquad (1)$$

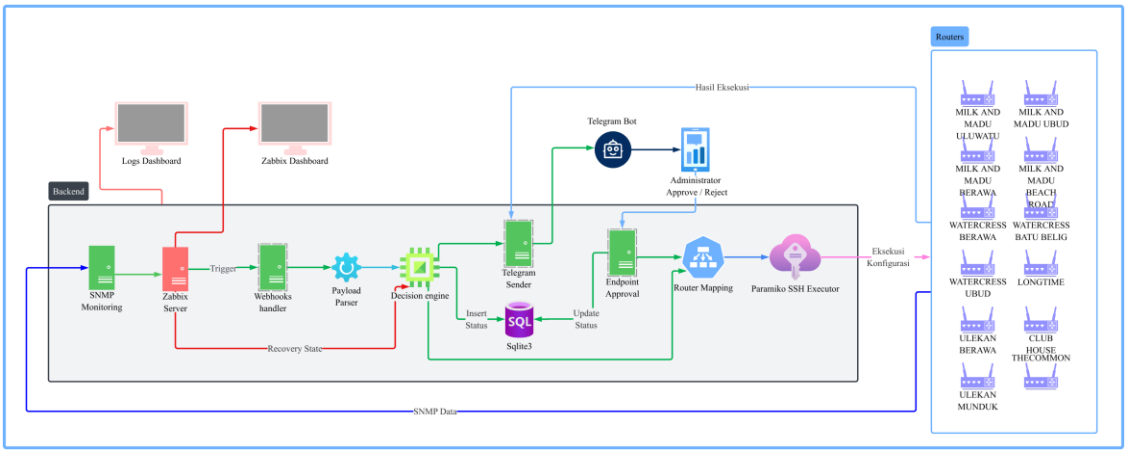
Dalam perhitungan tersebut, ΣR menunjukkan jumlah skor respons yang diperoleh dari seluruh penilaian, sedangkan N merupakan skor maksimum yang dapat dicapai berdasarkan instrumen penilaian yang digunakan. Hasil perhitungan persentase UAT digunakan untuk mengetahui tingkat penerimaan pengguna terhadap sistem yang dikembangkan. Nilai persentase yang lebih tinggi menunjukkan tingkat penerimaan pengguna yang lebih baik terhadap sistem.

Selanjutnya, *Performance Testing* digunakan untuk mengukur efisiensi dan keberhasilan proses otomasi dengan membandingkan waktu konfigurasi bandwidth secara manual dengan waktu konfigurasi menggunakan sistem pada *workflow problem* dan *workflow recovery*. Hasil pengujian tersebut menjadi dasar dalam mengevaluasi peningkatan efisiensi waktu setelah penerapan *network automation*.

3. HASIL DAN PEMBAHASAN

3.1 Implementasi Sistem Monitoring dan *Network Automation*

Penelitian ini menghasilkan sistem monitoring jaringan yang terintegrasi dengan mekanisme *network automation* untuk mendukung pengelolaan bandwidth pada lingkungan jaringan multi-cabang Milk & Madu Group. Sistem dikembangkan dengan mengintegrasikan Zabbix sebagai platform monitoring, Flask sebagai *backend* pemrosesan data, Telegram Bot sebagai media komunikasi administrator, SQLite sebagai basis data, serta Paramiko sebagai penghubung antara *backend* dan perangkat MikroTik melalui koneksi Secure Shell (SSH). Integrasi beberapa komponen tersebut memungkinkan proses monitoring dan pengelolaan jaringan dilakukan secara terpusat dalam satu sistem. Alur integrasi antara komponen monitoring, pemrosesan data, persetujuan administrator, dan eksekusi konfigurasi ditunjukkan pada Gambar 2.



Gambar 2. Arsitektur Sistem Monitoring dan Network Automation

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

Berdasarkan Gambar 2, proses dimulai dari SNMP Monitoring yang mengumpulkan data kondisi perangkat jaringan dan meneruskannya ke Zabbix Server. Ketika kondisi yang sesuai dengan aturan *trigger* terdeteksi, Zabbix meneruskan informasi kejadian melalui Webhook Handler. Data *payload* kemudian diproses oleh Payload Parser dan diteruskan ke Decision Engine untuk menentukan tindakan berdasarkan kondisi jaringan. Informasi tindakan yang dihasilkan selanjutnya dikirim melalui Telegram Sender kepada administrator untuk memperoleh persetujuan.

Implementasi monitoring dilakukan pada 12 perangkat router yang berada di berbagai cabang Milk & Madu Group. Seluruh perangkat terdaftar sebagai *host* pada Zabbix Server sehingga kondisi jaringan dapat dipantau secara terpusat melalui satu *dashboard*. Parameter yang dipantau meliputi penggunaan bandwidth, utilisasi CPU, penggunaan memori, status antarmuka jaringan, dan ketersediaan perangkat. Tampilan dashboard monitoring jaringan pada Zabbix ditunjukkan pada Gambar 3.

Gambar 3. Dashboard Monitoring Jaringan Seluruh Cabang pada Zabbix

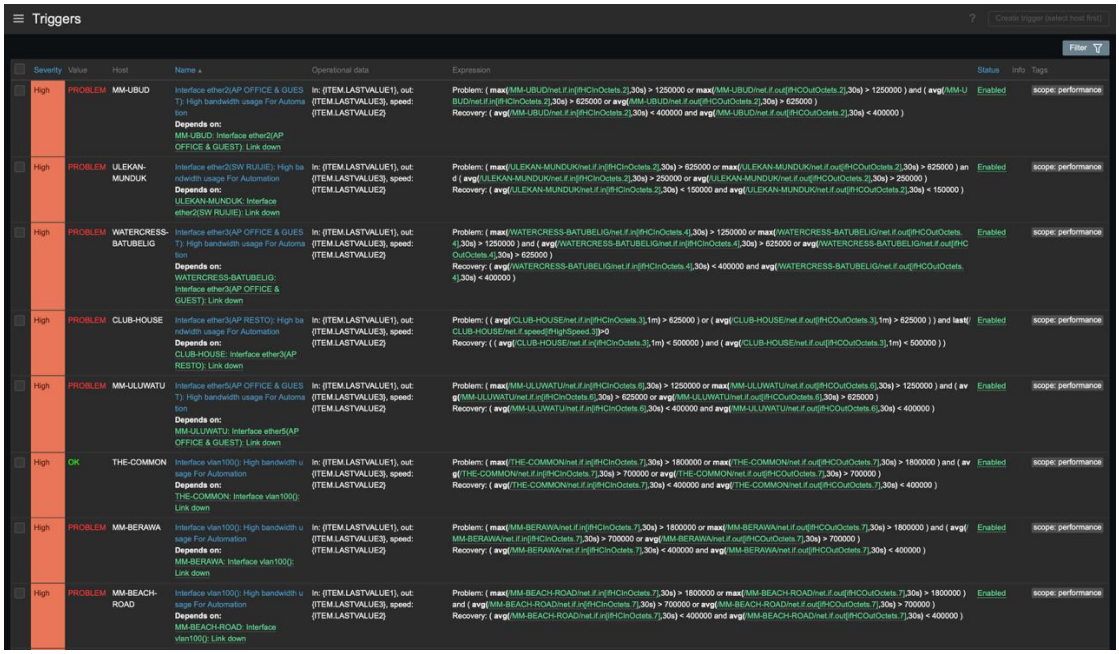
Gambar 3 menunjukkan dashboard monitoring Zabbix yang digunakan untuk memantau kondisi jaringan pada seluruh cabang Milk & Madu Group secara terpusat. Dashboard menyajikan informasi status perangkat, *uptime*, utilisasi CPU, serta trafik jaringan dari masing-masing perangkat secara *real-time*. Penyajian informasi dalam satu tampilan membantu administrator memperoleh gambaran kondisi jaringan tanpa harus melakukan pemeriksaan pada setiap perangkat secara langsung. Dengan demikian, dashboard dapat mendukung proses pemantauan dan identifikasi kondisi jaringan secara lebih efisien.

Hasil implementasi menunjukkan bahwa administrator dapat mengakses informasi kondisi jaringan secara *real-time* tanpa harus melakukan pengecekan pada setiap perangkat secara langsung. Kemampuan ini mempermudah proses pemantauan sekaligus mempercepat identifikasi permasalahan yang terjadi pada jaringan. Temuan tersebut mendukung hasil penelitian Guo et al. [20] yang menyatakan bahwa Zabbix efektif digunakan untuk monitoring jaringan secara terpusat pada lingkungan berskala besar. Selain itu, pemanfaatan dashboard monitoring membantu penyajian informasi secara lebih terstruktur sehingga mendukung proses pengambilan keputusan oleh administrator [5].

Integrasi antara monitoring dan *automation* diwujudkan melalui mekanisme *trigger* dan *webhook* pada Zabbix. Ketika parameter jaringan mencapai ambang batas tertentu, Zabbix menghasilkan *trigger* dan mengirimkan data kejadian ke *backend* Flask dalam format JSON. *Backend* kemudian menganalisis kondisi jaringan dan menghasilkan rekomendasi tindakan berdasarkan aturan yang telah ditentukan. Hasil analisis tersebut diteruskan kepada administrator melalui Telegram Bot sebagai mekanisme persetujuan sebelum konfigurasi dijalankan. Konfigurasi *trigger* yang digunakan sebagai pemicu *workflow automation* pada Zabbix ditunjukkan pada Gambar 4.

Gambar 4 menunjukkan konfigurasi *trigger* pada Zabbix yang digunakan untuk mendeteksi kondisi penggunaan bandwidth pada perangkat jaringan. Setiap *trigger* memiliki kondisi tertentu yang menentukan terjadinya status *problem* atau *recovery*. Ketika kondisi *problem* terpenuhi, Zabbix meneruskan informasi kejadian melalui mekanisme *webhook* untuk diproses oleh sistem otomasi. Ketika kondisi kembali normal, status *recovery* diteruskan melalui alur yang sama untuk menjalankan proses pemulihan konfigurasi.

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

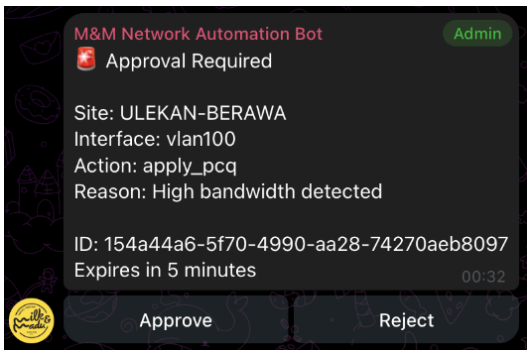


Gambar 4. Implementasi Trigger Network Automation pada Zabbix

Hasil implementasi menunjukkan bahwa integrasi antar komponen sistem mampu membentuk alur kerja yang terpusat mulai dari proses monitoring, deteksi permasalahan, pengambilan keputusan, hingga konfigurasi perangkat jaringan. Temuan ini memperlihatkan bahwa sistem monitoring tidak hanya berfungsi sebagai alat observasi, tetapi juga dapat dimanfaatkan sebagai sumber informasi untuk mendukung proses otomasi jaringan secara *real-time*. Integrasi monitoring dan otomasi yang diterapkan pada penelitian ini menerapkan konsep *event-driven automation*, di mana tindakan otomatis dijalankan berdasarkan kondisi yang terdeteksi oleh sistem monitoring. Melalui pendekatan tersebut, informasi hasil monitoring dapat dimanfaatkan secara langsung sebagai dasar pengambilan tindakan pada jaringan. Hasil penelitian ini mendukung temuan Santyadiputra et al. [21] yang menyatakan bahwa otomasi jaringan mampu meningkatkan efisiensi administrasi dengan mengurangi pekerjaan manual yang dilakukan secara berulang oleh administrator.

3.2 Implementasi Otomasi Manajemen Bandwidth

Mekanisme otomasi yang dikembangkan pada penelitian ini difokuskan pada pengelolaan bandwidth menggunakan metode *Per Connection Queue* (PCQ) pada perangkat MikroTik. Ketika sistem mendeteksi kondisi tertentu berdasarkan hasil monitoring, *backend* menghasilkan rekomendasi konfigurasi yang dikirimkan kepada administrator melalui Telegram Bot. Administrator kemudian dapat memberikan persetujuan sebelum tindakan konfigurasi dijalankan. Setelah memperoleh persetujuan, sistem secara otomatis menerapkan konfigurasi pada perangkat MikroTik melalui koneksi SSH menggunakan *library* Paramiko. Mekanisme persetujuan konfigurasi melalui Telegram Bot ditunjukkan pada Gambar 5.



Gambar 5. Implementasi Persetujuan Konfigurasi melalui Telegram Bot

Gambar 5 menunjukkan notifikasi *Approval Required* yang dikirimkan kepada administrator ketika sistem mendeteksi kondisi *high bandwidth*. Notifikasi memuat informasi *site*, *interface*, tindakan yang direkomendasikan, alasan konfigurasi, serta batas waktu persetujuan. Administrator dapat memilih *Approve* untuk mengizinkan eksekusi konfigurasi atau *Reject* untuk menolak tindakan yang diusulkan. Mekanisme tersebut menempatkan administrator sebagai pengendali keputusan sebelum konfigurasi diterapkan pada perangkat MikroTik.

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

Penggunaan metode PCQ memungkinkan bandwidth didistribusikan secara lebih adil kepada setiap pengguna yang aktif di jaringan. Hal ini penting mengingat setiap cabang Milk & Madu Group memiliki kebutuhan operasional dan pola penggunaan jaringan yang berbeda-beda. Berdasarkan hasil implementasi, metode PCQ mampu membantu menjaga pemerataan penggunaan bandwidth sehingga kualitas layanan jaringan tetap terjaga meskipun terjadi peningkatan jumlah pengguna atau trafik pada waktu tertentu. Hasil tersebut sejalan dengan penelitian Dzikrilloh Anwar dan Akbar [13] yang menunjukkan bahwa metode PCQ efektif dalam membagi bandwidth secara merata pada setiap koneksi aktif. Hasil serupa juga dilaporkan oleh Auriga et al. [12] yang menemukan bahwa metode PCQ memberikan performa yang baik dalam pengelolaan bandwidth pada jaringan berbasis MikroTik.

Penerapan mekanisme *Human-in-the-Loop* melalui Telegram Bot menjadi salah satu aspek penting dalam sistem yang dikembangkan. Pada mekanisme ini, administrator tetap dilibatkan dalam proses pengambilan keputusan sebelum konfigurasi dijalankan sehingga tindakan yang dilakukan sistem dapat dikontrol dan diverifikasi terlebih dahulu. Hasil implementasi menunjukkan bahwa pendekatan tersebut mampu meningkatkan efisiensi pengelolaan jaringan tanpa mengurangi aspek kontrol dan keamanan yang diperlukan dalam administrasi jaringan. Temuan ini menunjukkan bahwa otomasi tidak selalu harus menggantikan peran administrator, melainkan dapat berfungsi sebagai pendukung dalam proses pengambilan keputusan dan eksekusi konfigurasi jaringan. Dengan demikian, sistem tidak menggantikan peran administrator, melainkan berfungsi sebagai pendukung pengambilan keputusan (*decision support system*) yang membantu administrator merespons kondisi jaringan secara lebih cepat dan konsisten.

3.3 Pengujian Sistem

3.3.1 Black Box Testing

Pengujian fungsional dilakukan menggunakan metode *Black Box Testing* untuk memastikan seluruh fitur sistem dapat berjalan sesuai dengan kebutuhan yang telah ditetapkan. Pengujian mencakup fungsi monitoring jaringan, pengelolaan host, deteksi *trigger*, pengiriman *webhook*, komunikasi Telegram Bot, proses persetujuan administrator, eksekusi konfigurasi bandwidth, proses *recovery*, serta pencatatan aktivitas sistem.

Berdasarkan hasil pengujian, seluruh skenario yang diuji berhasil dijalankan sesuai dengan keluaran yang diharapkan. Dashboard monitoring mampu menampilkan kondisi perangkat secara *real-time*, mekanisme *trigger* dapat mendeteksi kondisi jaringan berdasarkan parameter yang telah ditentukan, dan Telegram Bot berhasil mengirimkan notifikasi serta menerima respons dari administrator. Selain itu, proses konfigurasi otomatis pada perangkat MikroTik melalui koneksi SSH juga dapat dijalankan tanpa kendala.

Hasil tersebut menunjukkan bahwa integrasi antara Zabbix, backend Flask, Telegram Bot, database SQLite, dan perangkat MikroTik telah berjalan dengan baik sebagai satu kesatuan sistem. Keberhasilan seluruh skenario pengujian mengindikasikan bahwa kebutuhan fungsional yang dirancang pada tahap analisis telah berhasil diimplementasikan. Temuan ini mendukung penelitian Amalia et al. [24] yang menyatakan bahwa *Black Box Testing* efektif digunakan untuk memverifikasi kesesuaian fungsi sistem berdasarkan keluaran yang dihasilkan. Hasil penelitian ini juga sejalan dengan Putri et al. [25] yang menunjukkan bahwa keberhasilan seluruh skenario pengujian merupakan indikator bahwa sistem telah memenuhi kebutuhan dan fungsi yang diharapkan pengguna.

3.3.2 User Acceptance Test (UAT)

Pengujian penerimaan pengguna (*User Acceptance Test* atau UAT) dilakukan dengan melibatkan dua staf Teknologi Informasi Milk & Madu Group, yaitu IT Manager dan IT Support. Evaluasi difokuskan pada aspek kemudahan penggunaan, kejelasan informasi yang disajikan sistem, keandalan fungsi monitoring, efektivitas mekanisme otomasi, serta kesesuaian sistem dengan kebutuhan operasional perusahaan.

Hasil pengujian menunjukkan tingkat penerimaan pengguna sebesar 96% yang termasuk dalam kategori Sangat Baik. Nilai tersebut menunjukkan bahwa pengguna menilai sistem mampu membantu proses monitoring dan pengelolaan jaringan secara lebih efektif dibandingkan mekanisme yang sebelumnya digunakan.

Tingginya tingkat penerimaan pengguna menunjukkan bahwa solusi yang dikembangkan telah sesuai dengan kebutuhan operasional perusahaan. Selain mendukung proses monitoring secara terpusat, sistem yang dikembangkan juga membantu administrator dalam melakukan konfigurasi jaringan dengan lebih cepat dan terstruktur. Integrasi antara monitoring dan otomasi memungkinkan respons terhadap perubahan kondisi jaringan dilakukan secara lebih efisien dibandingkan pendekatan manual yang sebelumnya digunakan. Hasil ini sejalan dengan penelitian Aliyah et al. [26] yang menyatakan bahwa nilai UAT yang tinggi menunjukkan sistem mampu memenuhi kebutuhan dan harapan pengguna. Dengan demikian, sistem yang dikembangkan tidak hanya layak digunakan dari sisi teknis, tetapi juga memiliki potensi untuk diterapkan pada perusahaan lain yang memiliki karakteristik jaringan multi-cabang dan keterbatasan sumber daya administrator.

3.3.3 Performance Testing

Pengujian performa dilakukan dengan membandingkan waktu yang diperlukan untuk melakukan konfigurasi bandwidth secara manual dan otomatis pada kondisi *workflow problem* dan *workflow recovery*. Tingkat efisiensi waktu dihitung menggunakan persamaan (2):

$$\text{Efisiensi Waktu}(\%) = \left(\frac{T_{manual} - T_{otomatis}}{T_{manual}}\right) \times 100\% \qquad (2)$$

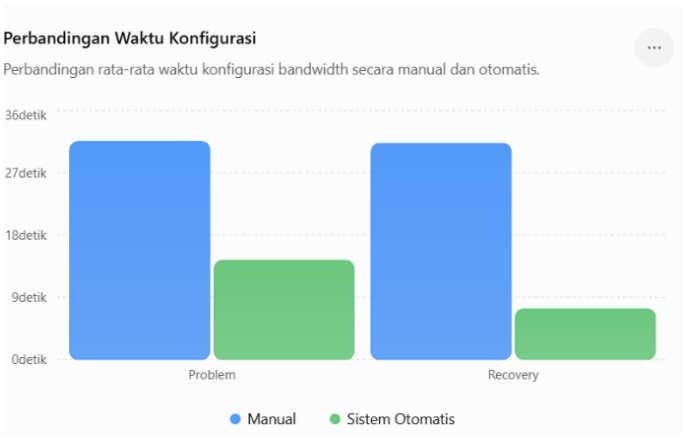
Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

Dalam perhitungan tersebut, T_{manual} menunjukkan durasi yang diperlukan untuk melakukan konfigurasi tanpa bantuan sistem, sedangkan $T_{otomatis}$ menunjukkan durasi yang diperlukan ketika proses konfigurasi dilakukan dengan bantuan sistem. Hasil pengujian waktu konfigurasi disajikan pada Tabel 1.

Tabel 1. Ringkasan Hasil Pengujian Efisiensi Waktu

Workflow	Rata-rata Manual (detik)	Rata-rata Sistem (detik)
Problem	31,51	14,31
Recovery	31,23	07,32

Untuk memvisualisasikan perbedaan waktu konfigurasi antara metode manual dan sistem pada masing-masing *workflow*, perbandingan rata-rata waktu konfigurasi disajikan pada Gambar 6.



Gambar 6. Perbandingan Waktu Konfigurasi Manual dan Otomatis

Berdasarkan Tabel 1 dan Gambar 6, sistem otomatis membutuhkan waktu yang lebih rendah dibandingkan metode manual pada kedua *workflow*. Pada *workflow problem*, rata-rata waktu konfigurasi secara manual sebesar 31,51 detik, sedangkan sistem otomatis membutuhkan 14,31 detik. Pada *workflow recovery*, rata-rata waktu konfigurasi manual sebesar 31,23 detik, sedangkan sistem otomatis membutuhkan 7,32 detik.

Perbedaan waktu tersebut menunjukkan bahwa penerapan *network automation* mampu meningkatkan efisiensi pengelolaan jaringan. Pada metode manual, administrator harus mengidentifikasi perangkat, mengakses router, dan menjalankan konfigurasi secara langsung. Sebaliknya, sistem otomatis mampu melakukan sebagian besar proses tersebut secara mandiri setelah memperoleh persetujuan administrator. Hasil penelitian ini mendukung temuan Santyadiputra et al. [21] yang menunjukkan bahwa otomasi jaringan mampu mengurangi aktivitas administratif yang dilakukan secara manual. Temuan ini juga sejalan dengan Mahant dan Kalapparambath [16] yang menyatakan bahwa *network automation* dapat meningkatkan efisiensi operasional dan mempercepat proses pengelolaan jaringan. Hasil pengujian tersebut menunjukkan bahwa penerapan network automation memberikan peningkatan efisiensi pada proses konfigurasi bandwidth. Dengan demikian, integrasi monitoring dan otomasi jaringan mampu meningkatkan efisiensi pengelolaan jaringan sekaligus mengurangi beban kerja administrator pada lingkungan jaringan multi-cabang.

Untuk mengetahui konsistensi eksekusi *workflow* yang dijalankan sistem, dilakukan pengujian tingkat keberhasilan pada *workflow problem* dan *workflow recovery*. Hasil pengujian tingkat keberhasilan disajikan pada Tabel 2.

Tabel 2. Hasil Pengujian Tingkat Keberhasilan

Workflow	Jumlah Pengujian	Berhasil	Gagal	Tingkat Keberhasilan
Problem	60	58	2	96,67%
Recovery	60	59	1	98,33%

Untuk memvisualisasikan perbandingan tingkat keberhasilan pada kedua *workflow*, persentase keberhasilan eksekusi sistem disajikan pada Gambar 7.

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

Gambar 7. Tingkat Keberhasilan Workflow Otomasi

Berdasarkan Tabel 2 dan Gambar 7, sistem berhasil menjalankan 58 dari 60 pengujian pada *workflow problem* dengan tingkat keberhasilan sebesar 96,67%, sedangkan pada *workflow recovery* sistem berhasil menjalankan 59 dari 60 pengujian dengan tingkat keberhasilan sebesar 98,33%. Hasil tersebut menunjukkan bahwa proses otomasi tidak hanya mampu meningkatkan efisiensi waktu konfigurasi, tetapi juga memiliki tingkat keandalan yang tinggi dalam pelaksanaannya. Konsistensi keberhasilan pada kedua *workflow* mengindikasikan bahwa integrasi antara Zabbix, backend Flask, Telegram Bot, dan perangkat MikroTik dapat bekerja dengan baik dalam mendukung proses monitoring dan pengelolaan jaringan secara otomatis. Temuan ini memperlihatkan bahwa sistem yang dikembangkan mampu memberikan keseimbangan antara efisiensi dan reliabilitas, sehingga berpotensi diterapkan pada lingkungan jaringan multi-cabang yang membutuhkan respons cepat terhadap perubahan kondisi jaringan.

3.4 Pembahasan

Hasil penelitian menunjukkan bahwa integrasi sistem monitoring jaringan berbasis Zabbix dengan *network automation* menggunakan Python dapat mendukung pengelolaan bandwidth pada lingkungan jaringan multi-cabang. Sistem yang dikembangkan menghubungkan proses monitoring, deteksi kondisi jaringan, pengambilan keputusan, persetujuan administrator, hingga eksekusi konfigurasi pada perangkat MikroTik dalam satu alur yang terintegrasi. Integrasi tersebut memberikan perbedaan dibandingkan mekanisme pengelolaan jaringan sebelumnya yang masih mengandalkan pemantauan dan konfigurasi secara manual pada masing-masing perangkat. Melalui pendekatan yang dikembangkan, informasi kondisi jaringan yang diperoleh dari proses monitoring dapat digunakan sebagai dasar untuk menentukan tindakan konfigurasi tanpa menghilangkan keterlibatan administrator dalam proses pengambilan keputusan.

Dari sisi monitoring jaringan, penggunaan Zabbix memungkinkan kondisi perangkat pada 12 cabang dipantau secara terpusat melalui satu dashboard. Informasi seperti status perangkat, *uptime*, utilisasi CPU, serta trafik jaringan dapat diamati tanpa administrator harus melakukan pemeriksaan secara langsung pada setiap router. Hasil tersebut sejalan dengan penelitian Guo et al. [20] yang memanfaatkan Zabbix sebagai platform monitoring untuk mendukung pengelolaan jaringan berskala besar. Kesamaan tersebut menunjukkan bahwa Zabbix dapat digunakan sebagai dasar pemantauan terpusat pada lingkungan jaringan dengan jumlah perangkat yang relatif banyak. Namun, penelitian ini memperluas pemanfaatan monitoring tersebut dengan menjadikan informasi hasil monitoring sebagai pemicu proses *network automation*. Dengan demikian, fungsi monitoring tidak berhenti pada penyajian informasi kondisi jaringan, tetapi dilanjutkan dengan proses pengambilan tindakan berdasarkan kondisi yang terdeteksi.

Pendekatan tersebut juga dapat dibandingkan dengan penelitian Fachrurrozi et al. [19] yang mengembangkan sistem monitoring jaringan berbasis LibreNMS dengan integrasi Telegram dan email sebagai media notifikasi. Kedua penelitian sama-sama memanfaatkan sistem monitoring untuk membantu administrator memperoleh informasi mengenai kondisi jaringan secara lebih cepat. Perbedaannya terletak pada fungsi informasi yang dihasilkan oleh sistem. Pada penelitian Fachrurrozi et al. [19], informasi monitoring digunakan terutama untuk memberikan notifikasi kepada administrator, sedangkan penelitian ini memanfaatkan informasi tersebut sebagai bagian dari *workflow automation*. Ketika kondisi tertentu terdeteksi oleh *trigger* Zabbix, informasi kejadian diteruskan melalui *webhook* ke backend Flask untuk diproses dan menghasilkan rekomendasi tindakan. Dengan mekanisme tersebut, sistem yang dikembangkan tidak hanya memberikan informasi kepada administrator, tetapi juga mempersiapkan tindakan konfigurasi berdasarkan kondisi jaringan yang terdeteksi.

Integrasi tersebut menunjukkan karakteristik *event-driven automation*, yaitu proses otomasi dijalankan berdasarkan kejadian atau kondisi tertentu yang dihasilkan oleh sistem monitoring. Pada penelitian ini, perubahan status *problem* dan *recovery* pada Zabbix menjadi dasar untuk menjalankan *workflow* otomasi. Pendekatan tersebut memiliki kesesuaian dengan penelitian Santyadiputra et al. [21] yang menunjukkan bahwa otomasi dapat membantu meningkatkan efisiensi administrasi jaringan dengan mengurangi pekerjaan manual yang dilakukan secara berulang. Perbedaannya, penelitian ini menghubungkan proses otomasi secara langsung dengan hasil monitoring jaringan dan

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

menerapkannya pada pengelolaan bandwidth perangkat MikroTik. Integrasi tersebut memungkinkan perubahan kondisi jaringan menjadi bagian dari alur konfigurasi, sehingga administrator tidak perlu memulai seluruh proses konfigurasi secara manual ketika kondisi yang telah ditentukan terdeteksi.

Pada aspek pengelolaan bandwidth, penelitian ini menggunakan metode *Per Connection Queue* (PCQ) untuk membantu mendistribusikan bandwidth kepada koneksi aktif pada perangkat MikroTik. Penggunaan PCQ memiliki kesesuaian dengan penelitian Dzikrilloh Anwar dan Akbar [13] serta Auriga et al. [12] yang menunjukkan bahwa PCQ dapat digunakan untuk mendukung pemerataan distribusi bandwidth pada jaringan berbasis MikroTik. Kesamaan hasil tersebut menunjukkan bahwa PCQ sesuai digunakan sebagai mekanisme pengelolaan bandwidth pada lingkungan jaringan yang membutuhkan distribusi bandwidth secara lebih teratur. Namun, kontribusi penelitian ini tidak hanya terletak pada penerapan PCQ, tetapi pada integrasinya dengan sistem monitoring dan *network automation*. Dengan demikian, konfigurasi bandwidth tidak hanya ditentukan melalui pengaturan yang dilakukan secara manual, tetapi dapat dipicu berdasarkan kondisi jaringan yang terdeteksi oleh sistem monitoring.

Jika dibandingkan dengan penelitian Dwipayoga et al. [14], penelitian ini memiliki kesamaan pada penerapan *network automation* untuk mendukung manajemen bandwidth pada perangkat MikroTik. Penelitian Dwipayoga et al. [14] menggunakan metode *Hierarchical Token Bucket* (HTB), sedangkan penelitian ini menggunakan PCQ. Perbedaan tersebut menunjukkan bahwa pendekatan otomasi dapat diterapkan bersama metode manajemen bandwidth yang berbeda sesuai kebutuhan konfigurasi jaringan. Selain perbedaan metode pengelolaan bandwidth, penelitian ini menambahkan integrasi dengan Zabbix sebagai sistem monitoring, *webhook* sebagai mekanisme komunikasi antarsistem, backend Flask sebagai pusat pemrosesan, serta Telegram Bot sebagai mekanisme persetujuan administrator. Oleh karena itu, proses otomasi yang dikembangkan tidak hanya berfokus pada eksekusi konfigurasi, tetapi membentuk alur yang menghubungkan monitoring, pengambilan keputusan, persetujuan, dan eksekusi.

Keberadaan mekanisme persetujuan administrator melalui Telegram Bot menjadi pembeda penting dalam sistem yang dikembangkan. Sistem tidak langsung menerapkan konfigurasi ketika kondisi *problem* terdeteksi, tetapi terlebih dahulu mengirimkan rekomendasi kepada administrator. Administrator dapat memberikan persetujuan atau menolak tindakan yang diusulkan sebelum konfigurasi diterapkan pada perangkat MikroTik. Pendekatan tersebut memberikan kontrol terhadap proses otomasi karena administrator tetap memiliki kewenangan dalam menentukan apakah perubahan konfigurasi dapat dijalankan. Dengan demikian, otomasi pada penelitian ini bersifat terkontrol dan tidak sepenuhnya menghilangkan peran administrator. Mekanisme tersebut juga mendukung penerapan otomasi pada lingkungan operasional yang membutuhkan kehati-hatian terhadap perubahan konfigurasi jaringan.

Berdasarkan hasil *performance testing*, penerapan *network automation* memberikan pengurangan waktu konfigurasi dibandingkan proses manual. Pada *workflow problem*, rata-rata waktu konfigurasi manual sebesar 31,51 detik, sedangkan sistem membutuhkan 14,31 detik. Pada *workflow recovery*, rata-rata waktu konfigurasi manual sebesar 31,23 detik, sedangkan sistem membutuhkan 7,32 detik. Hasil tersebut menunjukkan bahwa proses otomatis mampu mempercepat pelaksanaan konfigurasi pada kedua kondisi pengujian. Efisiensi tersebut berkaitan dengan berkurangnya aktivitas manual yang harus dilakukan administrator, seperti mengakses perangkat, menemukan konfigurasi yang diperlukan, dan menjalankan perubahan konfigurasi secara langsung. Temuan ini mendukung penelitian Mahant dan Kalapparambath [16] mengenai manfaat *network automation* dalam meningkatkan efisiensi operasional serta penelitian Santyadiputra et al. [21] mengenai pengurangan aktivitas administratif melalui otomasi.

Selain efisiensi waktu, hasil pengujian menunjukkan bahwa sistem memiliki tingkat keberhasilan yang tinggi dalam menjalankan *workflow*. Pada *workflow problem*, sistem berhasil menjalankan 58 dari 60 pengujian dengan tingkat keberhasilan 96,67%, sedangkan pada *workflow recovery* sistem berhasil menjalankan 59 dari 60 pengujian dengan tingkat keberhasilan 98,33%. Hasil tersebut menunjukkan bahwa integrasi antarkomponen tidak hanya mampu mempercepat proses konfigurasi, tetapi juga dapat menjalankan *workflow* secara konsisten pada sebagian besar pengujian. Perbedaan tingkat keberhasilan antara kedua *workflow* menunjukkan bahwa masih terdapat sejumlah kecil kondisi yang menyebabkan proses otomasi tidak berhasil, sehingga aspek penanganan kegagalan dan *error handling* masih menjadi bagian yang dapat dikembangkan lebih lanjut.

Hasil *User Acceptance Test* yang mencapai 96% juga memperlihatkan bahwa sistem memperoleh penerimaan yang tinggi dari pengguna yang terlibat dalam pengujian. Penilaian tersebut menunjukkan bahwa sistem dapat membantu pengguna dalam melakukan monitoring dan pengelolaan jaringan secara lebih terstruktur. Temuan tersebut sejalan dengan penelitian Aliyah et al. [26] yang menggunakan UAT untuk melihat kesesuaian sistem dengan kebutuhan pengguna. Dalam penelitian ini, tingginya nilai UAT tidak hanya berkaitan dengan fungsi monitoring, tetapi juga dengan kemudahan pengguna dalam menerima informasi, memberikan persetujuan, dan mengendalikan proses konfigurasi melalui Telegram Bot. Hal tersebut memperlihatkan bahwa integrasi beberapa komponen teknis tetap dapat digunakan dalam alur operasional yang melibatkan administrator sebagai pengambil keputusan.

Secara keseluruhan, hasil penelitian memperlihatkan adanya perbedaan pendekatan dibandingkan penelitian terdahulu yang umumnya membahas monitoring jaringan, notifikasi, otomasi administrasi, atau manajemen bandwidth secara terpisah. Penelitian [19] berfokus pada monitoring dan notifikasi, penelitian [20] pada monitoring menggunakan Zabbix, penelitian [21] pada otomasi administrasi jaringan, sedangkan penelitian [14] pada otomasi manajemen bandwidth MikroTik. Penelitian ini menggabungkan beberapa fungsi tersebut dalam satu *workflow*, yaitu monitoring kondisi jaringan melalui Zabbix, deteksi kondisi menggunakan *trigger*, pengiriman data melalui *webhook*, pemrosesan pada backend, persetujuan administrator melalui Telegram Bot, serta eksekusi konfigurasi bandwidth

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

pada MikroTik melalui SSH. Integrasi tersebut menjadi kontribusi utama penelitian dalam mendukung pengelolaan jaringan multi-cabang yang membutuhkan pemantauan terpusat dan respons konfigurasi yang lebih cepat.

4. KESIMPULAN

Penelitian ini berhasil mengimplementasikan sistem monitoring jaringan berbasis Zabbix yang terintegrasi dengan *network automation* untuk mendukung pengelolaan bandwidth pada lingkungan jaringan multi-cabang Milk & Madu Group. Sistem yang dikembangkan memanfaatkan Zabbix Server sebagai platform monitoring, backend berbasis Flask sebagai pusat pemrosesan data, Telegram Bot sebagai media interaksi dengan administrator, serta perangkat MikroTik yang dikonfigurasi melalui koneksi SSH menggunakan *library* Paramiko. Integrasi antar komponen tersebut memungkinkan proses monitoring, pengambilan keputusan, dan konfigurasi jaringan berjalan dalam satu alur yang saling terhubung. Hasil implementasi menunjukkan bahwa sistem mampu melakukan pemantauan kondisi jaringan secara *real-time* dan mendukung proses konfigurasi bandwidth secara otomatis berdasarkan kondisi yang terdeteksi oleh sistem monitoring. Seluruh fitur yang diuji melalui *Black Box Testing* berhasil berjalan sesuai dengan kebutuhan yang telah dirancang, sedangkan hasil *User Acceptance Test* (UAT) menunjukkan tingkat penerimaan pengguna sebesar 96% yang termasuk dalam kategori sangat baik. Pengujian performa menunjukkan efisiensi waktu sebesar 54,59% pada *workflow problem* dan 76,56% pada *workflow recovery*, dengan tingkat keberhasilan *workflow* masing-masing sebesar 96,67% dan 98,33%. Hasil tersebut menunjukkan bahwa integrasi monitoring dan otomasi dapat meningkatkan efisiensi pengelolaan jaringan sekaligus membantu mengurangi beban kerja administrator. Secara keseluruhan, penerapan *network monitoring* yang terintegrasi dengan *network automation* dapat menjadi solusi yang efektif bagi perusahaan dengan lingkungan jaringan multi-cabang dan jumlah administrator yang terbatas. Pengembangan selanjutnya dapat diarahkan pada penerapan mekanisme pengambilan keputusan yang lebih adaptif, perluasan cakupan otomasi konfigurasi jaringan, serta pemanfaatan teknologi analitik atau kecerdasan buatan untuk mendukung deteksi dan penanganan gangguan jaringan secara lebih proaktif.

REFERENCES

[1] G. A. J. Saskara, I. M. E. Listartha, I. P. S. D. Putra, and K. A. A. Kusuma, "Evaluasi Kualitas Jaringan Undiksha Harmoni dengan Menggunakan Metode Quality of Service Evaluation of the Quality of the Undiksha Harmoni Network Using the Quality of Service Method," *J. Teknol. dan Inf.*, vol. 14, no. 1, pp. 50–61, 2024, doi: 10.34010/jati.v14i1.

[2] R. F. Bari, A. Solehudin, and N. Heryana, "Analisis Quality of Service (QoS) Jaringan Internet Berbasis Wireless Local Area Network pada Layanan Indihome," *J. Ilm. Wahana Pendidik.*, vol. 8, no. 8, pp. 320–335, 2022, doi: https://doi.org/10.5281/zenodo.6820184.

[3] K. S. Pradipta, G. Arna, J. Saskara, B. Gede, and K. Yudistira, "Implementasi Kubernetes Cluster untuk High Availability dan Load Balancing SIAK Undiksha," *J. Artif. Intell. Digit. Bus.*, vol. 4, no. 4, pp. 12831–12840, 2026, doi: https://doi.org/10.31004/riggs.v4i4.5846.

[4] N. O. Miracle, "The Role of Network Monitoring and Analysis in Ensuring Optimal Network Performance," *Int. Res. J. Mod. Eng. Technol. Sci.*, vol. 6, no. 06, pp. 3009–3025, 2024, doi: https://www.doi.org/10.56726/IRJMETS59269.

[5] O. Y. Kovalenko and K. V Kuzniuk, "Computer network monitoring systems," *Teknol. Inf. dan Telekomun.*, no. 1, pp. 50–59, 2023, doi: 10.34121/1028-9763-2023-1-50-59.

[6] M. Diah, A. Devi, I. G. Ayu, A. Diatri, and I. K. Resika, "Dashboard Monitoring Alumni dengan Teknologi Business Intelligence pada Sistem Tracer Study Undiksha," *Inf. Syst. Emerg. Technol. J.*, vol. 4, no. 1, pp. 13–25, 2023, doi: 10.23887/insert.v4i1.58275.

[7] E. A. Katonova, J. Dzubak, and P. Fecilak, "Automated Monitoring of Network Infrastructures Based on the Zabbix Solution," *ICETA*, pp. 283–288, 2023, doi: https://doi.org/10.1109/ICETA61311.2023.10344265.

[8] A. R. H. Velasco, E. E. G. Malla, R. D. C. C. Herrera, and F. D. M. Arévalo, "Real-time monitoring and alerting system using Zabbix and Grafana software for wireless Internet access service management," *CISTI*, pp. 1–6, 2023, doi: https://doi.org/10.23919/CISTI58278.2023.10211432.

[9] N. T. Uky, Y. P. K. Kelen, K. Fallo, and E. J. Blegur, "Implementasi Manajemen Bandwidth Jaringan menggunakan Metode Queue Tree pada SMA Negeri 1 Kefamenanu," *J. Inform. dan Tek. Elektro Terap.*, vol. 14, no. 1, 2019, doi: https://doi.org/10.23960/jitet.v14i1.8300.

[10] I. M. E. Listartha and G. A. J. Saskara, "Security Testing with Penetration Testing Execution Standard (PTES) Methods to Find Misconfigurations Vulnerabilities in Network Devices," *J. Elektro Luceat*, vol. 10, no. 2, 2024.

[11] B. A. Prakosa, Y. Afrianto, S. Agustiyan, and I. H. Setiadi, "Ingénierie des Systèmes d ' Information Evaluating Bandwidth Management Techniques on Mikrotik Routers : A Multiple Linear Regression Approach," *Ingénierie des Systèmes d'Information*, vol. 29, no. 4, pp. 1561–1572, 2024, doi: https://doi.org/10.18280/isi.290429.

[12] W. Auriga, Y. Yuhandri, and S. Sumijan, "Jurnal KomtekInfo Komparasi Algoritma Queue Type SFQ , RED , FIFO dan PCQ pada Jaringan Nirkabel Berbasis Router Mikrotik," *J. KomtekInfo*, vol. 11, no. 3, pp. 122–131, 2024, doi: https://doi.org/10.35134/komtekinfo.v11i3.550.

[13] I. D. Anwar and Y. Akbar, "Manajemen Bandwidth Jaringan dengan Metode Per Connection Queue (PCQ) Pada Mikrotik di Masterpiece Family Karaoke Tebet Abstrak," *J. Indones. Manaj. Inform. dan Komun.*, vol. 5, no. 3, pp. 3125–3137, 2024, doi: https://doi.org/10.35870/jimik.v5i3.986.

[14] D. Made, W. Dwipayoga, G. Arna, J. Saskara, and I. M. G. Sunarya, "Pengembangan Website Network Automation untuk Manajemen Bandwidth Mikrotik dengan Metode Hierarchical Token Bucket (HTB) di SMPN 8 Singaraja," vol. 14, no. 2, pp. 236–248, 2025, doi: 10.23887/karmapati.v14i2.94491.

Bulletin of Information System Research (BIOS)
Volume 4, No 3, Agustus 2026, Page: 202−213
ISSN 2963-2455 (media online)
DOI: doi.org/10.62866/bios.v4i3.403,
https://journal.grahamitra.id/index.php/bios

[15] I. M. I. Kesuma, G. A. J. Saskara, and I. G. P. Sindu, "Pengembangan Sistem Automasi Pembayaran Langganan Layanan Internet pada Wahyu Net menggunakan Metode Rapid Application Development," *Kumpul. Artik. Mhs. Pendidik. Tek. Inform.*, vol. 15, no. 2, pp. 582–593, 2026, doi: 10.23887/karmapati.v15i2.116181.
[16] K. Mahant and J. P. Kalapparambath, "Exploring the Role of Network Automation in Enterprise Network Scalability and Efficiency," *Educ. Adm. Theory Pract.*, vol. 30, no. 11, pp. 1543–1549, 2024, doi: 10.53555/kuey.v30i11.9561.
[17] A. M. Mazin, R. A. Rahman, M. Kassim, and A. R. Mahmud, "Performance analysis on network automation interaction with network devices using python," *ISCAIE*, pp. 360–366, 2021, doi: https://doi.org/10.1109/ISCAIE51753.2021.9431823.
[18] Z. Tariq, B. Zainab, and M. Z. Hussain, "Evaluating the Effectiveness and Resilience of SSL / TLS , HTTPS , IPSec , SSH , and WPA / WPA2 in Safeguarding Data Transmission," *J. Eng. Technol.*, vol. 1, no. 2, pp. 1–7, 2024, doi: 10.24312/ucp-jeit.01.02.144.
[19] N. R. Fachrurrozi, A. A. Wirabudi, and S. A. Rozano, "Design of network monitoring system based on LibreNMS using Line Notify , Telegram , and Email notification," *SINERGI*, vol. 27, no. 1, pp. 111–122, 2023, doi: http://doi.org/10.22441/sinergi.2023.1.013 Design.
[20] F. Guo, C. Chen, and K. Li, "Research on Zabbix Monitoring System for Large-scale Smart Campus Network from a Distributed Perspective," *J. Electr. Syst.*, vol. 20, no. 10, pp. 631–348, 2024.
[21] G. S. Santyadiputra, I. M. E. Listartha, and G. A. J. Saskara, "The effectiveness of Automatic Network Administration (ANA) in network automation simulation at Universitas Pendidikan Ganesha The effectiveness of Automatic Network Administration (ANA) in network automation simulation at Universitas Pendidikan Ganes," *J. Phys.*, vol. 1810, no. 1, p. 012028, 2021, doi: 10.1088/1742-6596/1810/1/012028.
[22] P. D. Sugiyono, *Metode Penelitian Kuantitatif, Kualitatif, dan R&D*, vol. 11, no. 1. 2023. [Online]. Available:http://scioteca.caf.com/bitstream/handle/123456789/1091/RED2017-Eng-8ene.pdf?sequence=12&isAllowed=y%0Ahttp://dx.doi.org/10.1016/j.regsciurbeco.2008.06.005%0Ahttps://www.researchgate.net/publication/305320484_SISTEM_PEMBETUNGAN_TERPUSAT_STRATEGI_MELESTARI
[23] A. Binuko Paksi, U. Hafidhoh, and S. Kariagil Bimonugroho, "Perbandingan Model Pengembangan Perangkat Lunak Untuk Proyek Tugas Akhir Program Vokasi Program Studi D3 Teknologi Informasi, Politeknik Negeri Madiun," *J. Masy. Inform.*, vol. 14, no. 1, pp. 70–79, 2023, doi: https://doi.org/10.14710/jmasif.14.1.52752.
[24] A. Amalia, S. Wanda, P. Hamidah, and T. Kristanto, "Pengujian Black Box Menggunakan Teknik Equivalence Partitions Pada Aplikasi E-Learning Berbasis Web," *Build. Informatics, Technol. Sci.*, vol. 3, no. 3, pp. 269–274, 2021, doi: 10.47065/bits.v3i3.1062.
[25] S. J. Putri *et al.*, "Analisis Komparasi pada Teknik Black Box Testing (Studi Kasus : Website Lars)," *J. Internet Softw. Eng.*, vol. 5, no. 1, pp. 23–28, 2024, doi: DOI: https://doi.org/10.22146/jise.v5i1.9446.
[26] Aliyah, N. Hartono, and A. A. Muin, "Penggunaan User Acceptance Testing (UAT) Pada Pengujian Sistem Informasi Pengelolaan Keuangan Dan Inventaris Barang," *J. Sains dan Teknol. Inf.*, vol. 3, no. 2, pp. 42–58, 2025, doi: https://doi.org/10.62951/switch.v3i1.330.