Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

Penerapan Ansible untuk Otomasi Konfigurasi dan Monitoring Perangkat Jaringan Multi-Vendor Pada Lingkungan Enterprise

Dimas Agung Sutrishna[1], Fandi Ali Mustika[2*], Febryo Ponco Sulistyo[3], Yuwan Jumaryadi[4]
[1,2,3]Teknik Informatika, Universitas Mercu Buana, Indonesia
[1]41519110005@student.mercubuana.ac.id, [2*]fandi.ali@mercubuana.ac.id,
[3]febryo.ponco@mercubuana.ac.id, [3]yuwan.jumaryadi@mercubuana.ac.id

Abstrak: Pengelolaan infrastruktur jaringan multi-vendor menghadapi perbedaan sistem operasi, sintaks konfigurasi, model data, dan mekanisme koneksi sehingga konfigurasi manual berisiko menimbulkan inkonsistensi serta menyulitkan verifikasi dan pemulihan. Penelitian ini bertujuan membangun prototipe Zero Touch Branch Deployment (ZTBD) berbasis Ansible yang mengintegrasikan otomasi konfigurasi dan monitoring pada perangkat MikroTik RouterOS, VyOS, dan FortiGate. Penelitian menggunakan pendekatan terapan dan rekayasa sistem dengan pengembangan prototipe secara iteratif pada lingkungan virtual VMware dan PNETLab. Ansible Control Node digunakan untuk menjalankan validasi konektivitas, backup pre-change, deployment, verification terhadap desired state, pengujian idempotency, controlled failure, dan state-aware rollback. Integrasi monitoring dilakukan dengan Zabbix melalui JSON-RPC API dan SNMPv2c. Evaluasi menggunakan pengujian black-box terhadap fungsi jaringan, hasil eksekusi playbook, kesesuaian konfigurasi, availability SNMP, serta fault recovery. Hasil pengujian menunjukkan seluruh 15 skenario memperoleh status PASS. Eksekusi ulang deployment pada ketiga vendor menghasilkan changed=0, sedangkan workflow end-to-end berakhir dengan failed=0 dan unreachable=0. Skenario controlled failure pada alias FortiGate berhasil memicu rescue, mengembalikan state sebelumnya, dan memverifikasi kondisi akhir. Zabbix berhasil mendaftarkan tiga perangkat tanpa duplikasi, melakukan polling SNMPv2c, mendeteksi gangguan SNMP MikroTik, dan memulihkan status availability setelah layanan diaktifkan kembali. Hasil ini menunjukkan bahwa ZTBD mampu menyediakan otomasi konfigurasi yang terpusat, konsisten, dapat diverifikasi, dan terintegrasi dengan monitoring dalam lingkup prototipe laboratorium.
Kata Kunci: Ansible; Network Automation; Multi-Vendor Network; Zabbix

Abstract: Managing a multi-vendor network infrastructure is challenging because network devices differ in operating systems, configuration syntax, data models, and connection mechanisms, making manual configuration prone to inconsistency and difficult verification and recovery. This study aims to develop an Ansible-based Zero Touch Branch Deployment (ZTBD) prototype that integrates configuration automation and monitoring for MikroTik RouterOS, VyOS, and FortiGate devices. The study applies an applied research and systems engineering approach with iterative prototyping in a virtualized VMware and PNETLab environment. Ansible Control Node performs connectivity and identity validation, pre-change

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

configuration backup, deployment, desired-state verification, idempotency testing, controlled failure, and state-aware rollback. Monitoring integration is implemented using Zabbix through its JSON-RPC API and SNMPv2c. Evaluation uses black-box testing covering network functions, playbook execution results, configuration conformity, SNMP availability, and fault recovery. The results show that all 15 test scenarios achieved PASS status. Repeated deployment on all three vendors produced changed=0, while the end-to-end workflow completed with failed=0 and unreachable=0. A controlled failure involving the FortiGate alias successfully triggered the rescue process, restored the previous state, and verified the final condition. Zabbix successfully registered three devices without duplication, performed SNMPv2c polling, detected an SNMP service disruption on MikroTik, and restored availability after the service was re-enabled. These findings indicate that ZTBD can provide centralized, consistent, verifiable, and monitoring-integrated network configuration automation within the scope of a laboratory prototype.
Keywords: Ansible; Network Automation; Multi-Vendor Network; Zabbix

1. PENDAHULUAN

Infrastruktur jaringan enterprise semakin sering dibangun dari perangkat dengan fungsi dan platform yang berbeda. Lingkungan multi-vendor memberikan fleksibilitas dalam pemilihan teknologi, tetapi juga meningkatkan kompleksitas operasional karena setiap platform dapat memiliki sintaks, model data, antarmuka, dan mekanisme koneksi yang berbeda. Kebutuhan otomasi menjadi semakin relevan karena organisasi perlu menjaga konsistensi konfigurasi, mempercepat perubahan, dan menyediakan visibilitas kondisi perangkat secara berkelanjutan [1], [2].

Pada pendekatan manual, administrator harus mengakses perangkat satu per satu, memahami perintah spesifik vendor, menerapkan perubahan, kemudian memeriksa kembali hasilnya. Pola tersebut meningkatkan risiko human error, konfigurasi yang tidak konsisten, serta kesulitan audit dan pemulihan. Studi sebelumnya menunjukkan bahwa Ansible dapat digunakan untuk otomasi routing, VLAN, DHCP, EIGRP, dan layanan jaringan lain, sedangkan pendekatan berbasis Python juga telah digunakan untuk konfigurasi massal perangkat [3], [4], [5], [6].

Penelitian terdahulu kemudian berkembang menuju pengelolaan jaringan heterogen, single source of truth, virtual network, backup konfigurasi, dan monitoring. Mulyana dan Fakih [7] menunjukkan manfaat sumber konfigurasi terpusat pada lingkungan heterogen, sementara Nugroho dan Pujiarto [8] memperlihatkan pengelolaan beberapa router melalui antarmuka terpusat. Pada sisi monitoring, Zabbix berbasis SNMP telah digunakan untuk memberikan informasi kondisi perangkat secara terpusat [9], dan Infrastructure as Code berbasis Ansible telah diterapkan untuk otomasi keamanan serta monitoring infrastruktur [10]. Penelitian lain juga menunjukkan penggunaan Ansible pada virtual network dan SDN [11], [12], serta otomasi backup MikroTik [13], [14].

Meskipun demikian, fungsi-fungsi tersebut umumnya masih diuji dengan fokus yang terpisah. Survei zero-touch management menempatkan otomasi sebagai arah menuju pengurangan intervensi manual [1], sedangkan studi multi-vendor terbaru juga menunjukkan bahwa otomasi dapat diterapkan lintas platform [15]. Namun, berdasarkan sintesis penelitian pada sumber utama, masih terdapat integration gap berupa belum terintegrasinya pemeriksaan kondisi awal, backup pre-change, deployment multi-vendor, verification terhadap desired state, pembuktian idempotency, state-aware rollback, registrasi monitoring melalui API, serta pengujian fault dan recovery monitoring dalam satu workflow end-to-end [7], [10], [13], [15].

Fandi Ali Mustika: *Penulis Korespondensi

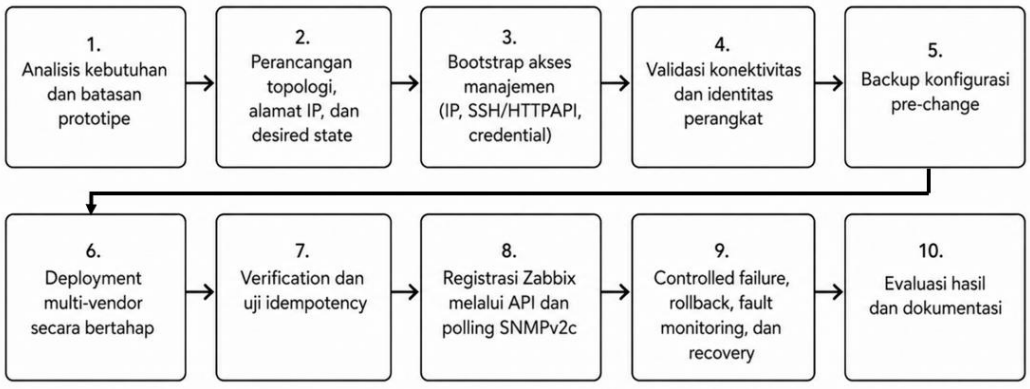
Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

Penelitian ini mengusulkan prototipe Zero Touch Branch Deployment (ZTBD) berbasis Ansible yang menyatukan siklus Bootstrap–Validate–Backup–Deploy–Verify–Idempotency–Zabbix Registration–Monitoring, dengan skenario terpisah Controlled Failure–State-Aware Rollback–Post-Rollback Verification. Kebaruan penelitian bukan terletak pada penggunaan Ansible atau Zabbix sebagai teknologi baru, melainkan pada integrasi lifecycle otomasi konfigurasi dan monitoring yang dapat diverifikasi pada tiga platform berbeda, yaitu MikroTik RouterOS, VyOS, dan FortiGate. Pendekatan ini mempertahankan mekanisme koneksi dan modul yang sesuai karakteristik tiap vendor, tetapi tetap dikendalikan dari satu control node [3], [7], [12], [13], [16].
Tujuan penelitian adalah membangun dan mengevaluasi prototipe ZTBD untuk: (1) mengotomasi konfigurasi tiga perangkat multi-vendor melalui satu Ansible Control Node; (2) menerapkan validasi, backup, deployment, verification, dan idempotency; (3) membuktikan controlled failure dan state-aware rollback; serta (4) mengintegrasikan registrasi host dan monitoring Zabbix melalui JSON-RPC API dan SNMPv2c. Kontribusi praktis penelitian adalah menyediakan pola workflow yang terpusat, dapat ditelusuri, dan dapat dikembangkan menuju standardisasi konfigurasi cabang enterprise, dengan batasan bahwa pengujian dilakukan pada lingkungan laboratorium virtual [9], [10], [17], [14].

2. METODE PENELITIAN

2.1. Tahapan Penelitian

Penelitian menggunakan pendekatan penelitian terapan dan rekayasa sistem dengan pengembangan prototipe secara iteratif. Tahapan dimulai dari analisis kebutuhan dan batasan, perancangan topologi serta desired state, penyediaan bootstrap management access, validasi konektivitas dan identitas, backup pre-change, deployment multi-vendor, verification, pengujian idempotency, integrasi Zabbix, controlled failure dan rollback, hingga evaluasi hasil yang dapat dilihat pada gambar 1. Alur tersebut dirancang agar setiap tahap menghasilkan artefak atau indikator yang dapat diverifikasi sebelum proses dilanjutkan ke tahap berikutnya [18]. Lingkungan virtual dipilih agar skenario dapat direproduksi tanpa memengaruhi jaringan produksi, sebagaimana praktik penggunaan simulator, emulator, dan testbed untuk penelitian jaringan [19].



Gambar 1. Alur Penelitian dan Pengembangan Prototipe ZTBD

2.2. Metode Pegembangan Prototipe

Prototipe dibangun secara bertahap dengan prinsip modular. Ansible Control Node menjadi pusat orkestrasi, sedangkan perangkat dikelompokkan dalam inventory

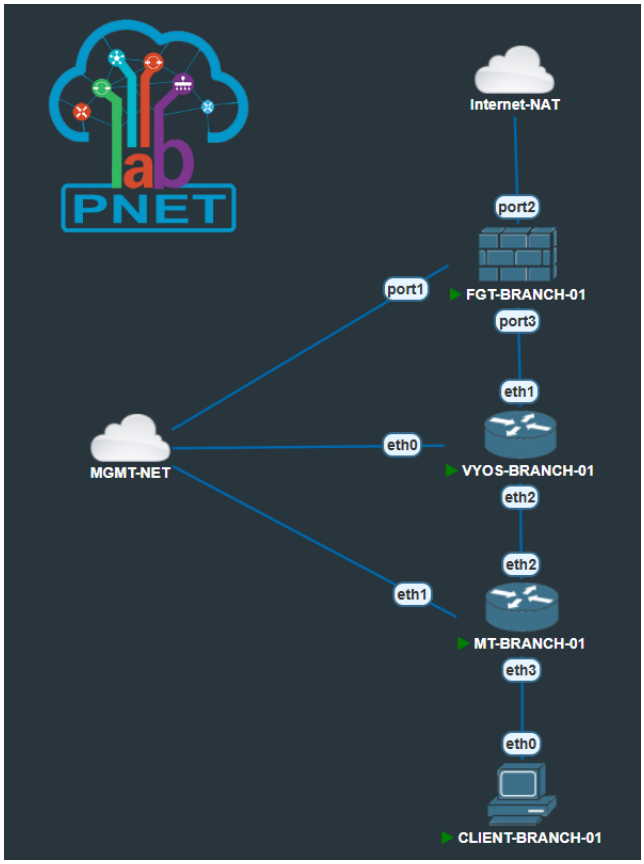
Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

berdasarkan vendor dan metode koneksinya. Desired state nonrahasia ditempatkan pada variable, sementara password perangkat, token FortiGate, SNMP community, dan token API Zabbix dipisahkan menggunakan Ansible Vault. MikroTik dan VyOS dikelola menggunakan network_cli berbasis SSH; FortiGate menggunakan HTTPAPI dan modul resource-specific. Pendekatan ini mengikuti kebutuhan pengelolaan Infrastructure as Code yang menekankan konsistensi, keterulangan, dan kontrol terhadap perubahan [20], [21].

2.3. Lingkungan Implementasi dan Arsitektur

Implementasi dilakukan pada VMware dan PNETLab. Ubuntu Server 24.04.4 LTS digunakan sebagai Ansible Control Node sekaligus Zabbix Server. PNETLab menjalankan MikroTik RouterOS CHR sebagai router cabang, VyOS sebagai router distribusi, FortiGate VM sebagai firewall/gateway, dan VPCS sebagai klien. Zabbix 7.0.28 digunakan untuk monitoring. Ansible berkomunikasi dengan MikroTik dan VyOS melalui SSH/network_cli dan FortiGate melalui HTTPAPI. Zabbix berkomunikasi dengan perangkat menggunakan SNMPv2c pada port 161, sedangkan registrasi host dilakukan melalui JSON-RPC API. Konfigurasi ini memisahkan management plane dari data plane dan memungkinkan pengujian konfigurasi serta observability secara terkontrol, bisa dilihat pada gambar 2 [9], [19].



Gambar 2. Topologi Jaringan ZTBD pada PNETLab.

Tabel 1. Lingkungan Implementasi Utama

Komponen	Spesifikasi/Versi	Fungsi
VM Ansible/Zabbix	2 vCPU; RAM 3,8 GiB; swap 3,8 GiB; disk 19 GiB	Control node dan monitoring server

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

Ubuntu Server	24.04.4 LTS	OS Ansible Control Node dan Zabbix
Ansible collections	Gfortinet.fortios 2.5.1; community.routeros 2.12.0; vyos.vyos 4.1.0	Orkestrasi konfigurasi multi-vendor
Zabbix	7.0.28	Monitoring melalui SNMPv2c
Net-SNMP	5.9.4.pre2	snmpget dan snmpwalk
PNETLab	Lingkungan emulasi	Menjalankan MikroTik, VyOS, FortiGate, dan VPCS

2.4. Perancangan Workflow dan Desired State

Workflow utama terdiri atas validasi, backup, deployment, verification, idempotency, registrasi Zabbix, dan monitoring. Desired state MikroTik mencakup identity, bridge/access port, VLAN 10, alamat IP, DHCP, DNS cache, default route, dan SNMP. Desired state VyOS mencakup hostname, interface transit, static route, dan SNMP. Desired state FortiGate mencakup alias, interface, route internal, address object, firewall policy dengan source NAT, dan SNMP. Pada MikroTik dan VyOS, idempotency diterapkan dengan pola check-before-change karena task command bersifat lebih imperatif; pada FortiGate digunakan modul resource-specific melalui HTTPAPI. Konsep desired state dan idempotency menjadi dasar agar eksekusi ulang tidak menghasilkan perubahan yang tidak diperlukan [17], [18].

Tabel 2. Skema Alamat IP Implementasi

Perangkat/Interface	Peran	Alamat IP	Keterangan
Ansible/Zabbix ens33	Management	192.168.188.130/24	Control node dan monitoring server
FortiGate port1	Management	192.168.188.201/24	HTTPAPI, SSH, HTTPS, SNMP
MikroTik ether1	Management	192.168.188.202/24	network_cli dan SNMP
VyOS eth0	Management	192.168.188.203/24	network_cli dan SNMP
FortiGate port3	Transit	10.10.1.1/30	Transit ke VyOS
VyOS eth1	Transit	10.10.1.2/30	Transit ke FortiGate
VyOS eth2	Transit	10.10.2.1/30	Gateway MikroTik
MikroTik ether2	Transit	10.10.2.2/30	Terhubung ke VyOS
MikroTik VLAN 10	Client Network	10.20.10.1/24	Gateway dan DNS klien
VPCS	Client	10.20.10.200/24	DHCP; gateway 10.20.10.1

2.5. Integrasi Monitoring dan Pengujian

Registrasi Zabbix dilakukan melalui API dengan terlebih dahulu memeriksa keberadaan host group dan host. Tiga host didaftarkan dengan template FortiGate by SNMP, Mikrotik by SNMP, dan Network Generic Device by SNMP untuk VyOS. Setelah registrasi, Zabbix melakukan polling SNMPv2c secara periodik. Pengujian black-box mencakup 15 skenario: validasi koneksi, backup, deployment tiga vendor, idempotency, verification, controlled failure, rollback, registrasi dan idempotency Zabbix, polling SNMP, fault detection, recovery, dan workflow end-to-end. Indikator evaluasi meliputi status PASS/FAIL, changed, failed, unreachable, skipped, rescued, kesesuaian actual state dengan desired state, keberhasilan fungsi jaringan, availability SNMP, dan keberhasilan workflow akhir [9], [22], [23].

3. HASIL DAN PEMBAHASAN

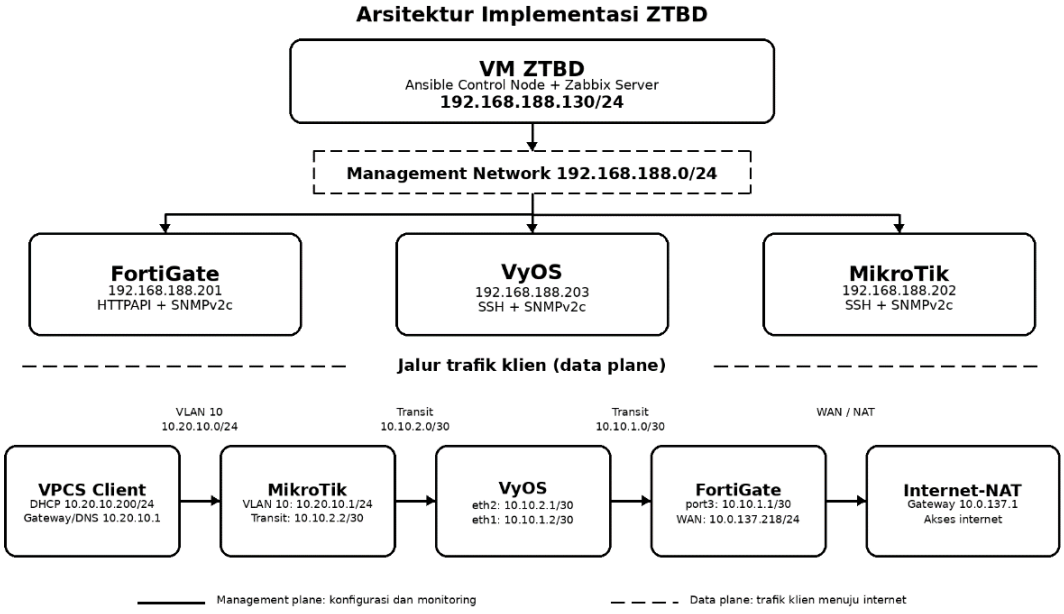
3.1. Implementasi Prototipe

Implementasi berhasil menempatkan Ansible Control Node sebagai pusat orkestrasi dan Zabbix sebagai pusat observability. Jaringan manajemen 192.168.188.0/24 menghubungkan control node dengan FortiGate, MikroTik, dan VyOS, sedangkan jalur data

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

membentuk urutan VPCS–MikroTik–VyOS–FortiGate–Internet-NAT. VM Ubuntu 24.04.4 LTS menggunakan 2 vCPU, RAM 3,8 GiB, swap 3,8 GiB, dan disk logis 19 GiB. Zabbix 7.0.28, Net-SNMP 5.9.4.pre2, serta collection fortinet.fortios 2.5.1, community.routeros 2.12.0, dan vyos.vyos 4.1.0 digunakan sesuai fungsi masing-masing.



Gambar 3. Arsitektur Implementasi ZTBD

3.2. Hasil Deployment, Verification, dan Idempotency

Deployment pada MikroTik menerapkan VLAN, DHCP, DNS, routing, dan SNMP; VyOS menerapkan interface transit, static route, dan SNMP; sedangkan FortiGate menerapkan route, address object, firewall policy dengan source NAT, dan SNMP. Pengujian verification pada MikroTik menghasilkan 13 task berhasil, VyOS 9 task, dan FortiGate 12 task, semuanya tanpa failed maupun unreachable. Eksekusi ulang deployment pada ketiga vendor menghasilkan changed=0. Pada MikroTik dan VyOS, beberapa task dilewati karena mekanisme check-before-change mengenali state yang telah sesuai; pada FortiGate, modul resource-specific mempertahankan konsistensi objek melalui HTTPAPI. Hasil tersebut menunjukkan bahwa desired state dapat dipertahankan melalui workflow yang berbeda sesuai karakteristik vendor.

Tabel 3. Ringkasan Deployment Ulang dan Verfikasi

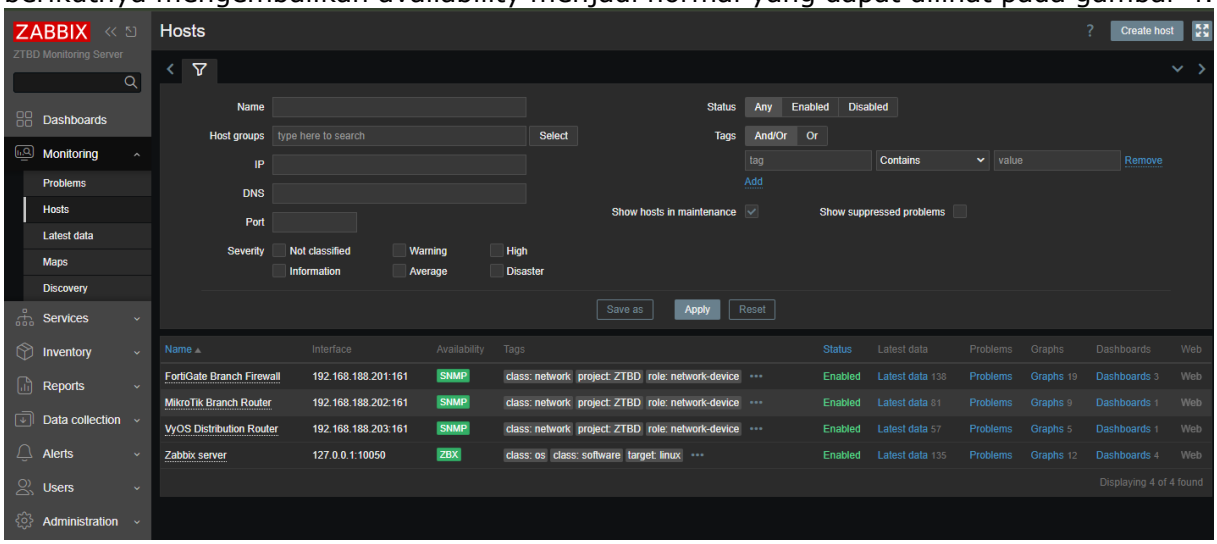
Komponen	Hasil Utama	Status
MikroTik	Deployment ulang changed=0; verification ok=13; failed=0; unreachable=0	PASS
VyOS	Deployment ulang changed=0; verification ok=9; failed=0; unreachable=0	PASS
FortiGate	Deployment ulang changed=0; verification ok=12; failed=0; unreachable=0	PASS
Control Node	Management dan SNMP check ok=5; changed=0; failed=0	PASS

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

3.3. Hasil Integrasi Monitoring dan Recovery

Integrasi Zabbix dilakukan melalui JSON-RPC API. Eksekusi pertama registrasi menghasilkan ok=15 dan changed=2, sedangkan eksekusi kedua menghasilkan ok=13, changed=0, dan skipped=2, sehingga host group dan host tidak dibuat ulang. Tiga perangkat terdaftar pada port SNMP 161 dan menunjukkan availability hijau. FortiGate menggunakan template FortiGate by SNMP, MikroTik menggunakan Mikrotik by SNMP, dan VyOS menggunakan Network Generic Device by SNMP. Saat layanan SNMP MikroTik dinonaktifkan sekitar lima menit, Zabbix mengubah availability menjadi unavailable sementara FortiGate dan VyOS tetap tersedia. Setelah SNMP diaktifkan kembali, polling berikutnya mengembalikan availability menjadi normal yang dapat dilihat pada gambar 4.



Gambar 4. Status Availability SNMP Tiga Perangkat pada Zabbix

Tabel 4. Registrasi dan Polling Zabbix

Host	IP:Port	Template	Availability	Latest Data
FGT-BRANCH-01	192.168.188.201:161	FortiGate by SNMP	SNMP hijau	138
MT-BRANCH-01	192.168.188.202:161	Mikrotik by SNMP	SNMP hijau	81
VYOS-BRANCH-01	192.168.188.203:161	Network Generic Device by SNMP	SNMP hijau	57

3.4. Hasil Controlled Failure dan State-Aware Rollback

Skenario rollback menggunakan alias FortiGate sebagai parameter representatif karena perubahan tersebut tidak memutus jalur management, routing, maupun firewall policy. Sistem terlebih dahulu menangkap alias awal, menerapkan alias sementara yang berbeda dari desired state, kemudian memicu ansible.builtin.fail untuk mengaktifkan blok rescue. Rescue mengembalikan alias ke nilai original_alias, sedangkan blok always melakukan verifikasi akhir. Hasil pengujian menunjukkan rescued=1 dan failed=0 serta pesan verifikasi bahwa state-aware rollback berhasil. Dengan demikian, rollback tidak sekadar mengembalikan nilai statis, tetapi menggunakan state sebelum perubahan sebagai dasar pemulihan.

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : https://doi.org/10.58602/itsecs.v4i2.390
Page 30-41

```
- name: Execute controlled failure scenario
  block:
      - name: Apply temporary incorrect FortiGate alias
        fortinet.fortios.fortios_system_global:
            system_global:
                  alias: "{{ failure_alias }}"
      - name: Trigger rollback workflow
            ansible.builtin.fail:
                  msg: "CONTROLLED FAILURE"
  rescue:
      - name: Restore FortiGate alias to previous state
            fortinet.fortios.fortios_system_global:
                  system_global:
                        alias: "{{ original_alias }}"
```

```
TASK [Verify controlled configuration deviation] ********************************************************************************
ok: [fgt-branch-01] => {
    "changed": false,
    "msg": "PASS - Controlled configuration deviation detected"
}

TASK [Trigger rollback workflow] ************************************************************************************************
fatal: [fgt-branch-01]: FAILED! => {"changed": false, "msg": "CONTROLLED FAILURE - FortiGate alias differs from desired state"}

TASK [Report controlled failure detection] **************************************************************************************
ok: [fgt-branch-01] => {
    "msg": "PASS - Controlled failure detected; rollback initiated"
}

TASK [Restore FortiGate alias to previous state] ********************************************************************************
changed: [fgt-branch-01]

TASK [Report rollback action] ***************************************************************************************************
ok: [fgt-branch-01] => {
    "msg": "PASS - FortiGate rollback command executed"
}

TASK [Read FortiGate state after rollback] **************************************************************************************
ok: [fgt-branch-01]

TASK [Verify FortiGate rollback result] *****************************************************************************************
ok: [fgt-branch-01] => {
    "changed": false,
    "msg": "PASS - FortiGate state-aware rollback verified"
}
```

Gambar 5. Controlled Failure dan State-Aware Rollback pada Fortigate

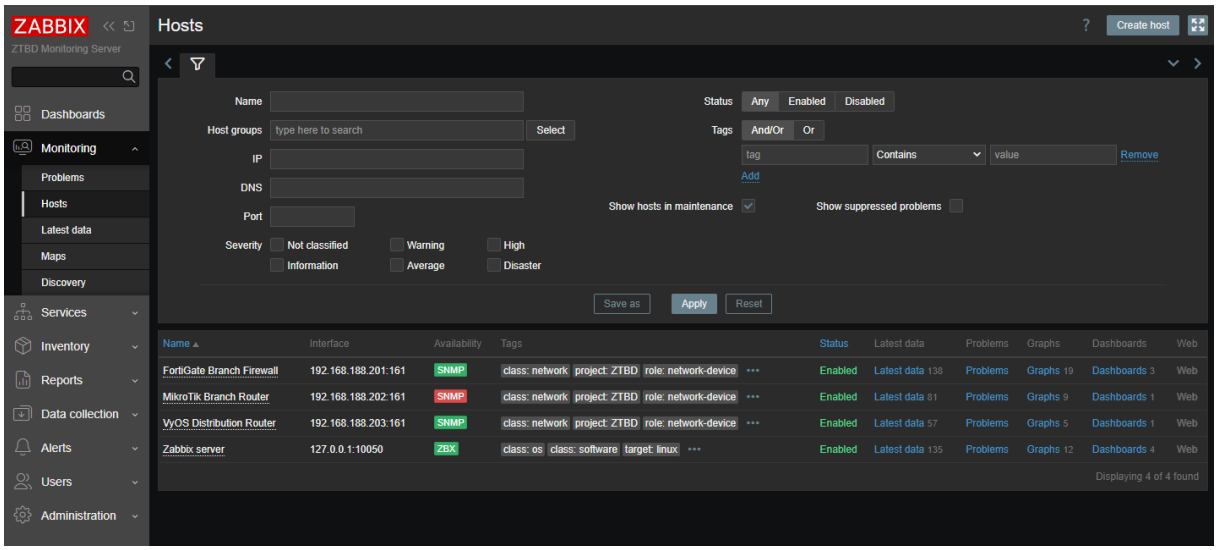
3.5. Hasil Pengujian Fault Monitoring

Kemampuan deteksi gangguan diuji dengan menonaktifkan SNMP pada MikroTik selama sekitar lima menit. Setelah beberapa siklus polling, indikator availability pada host MikroTik berubah merah atau unavailable, sedangkan FortiGate dan VyOS tetap hijau. Setelah SNMP diaktifkan kembali, snmpget dan playbook verification kembali berhasil, dan status MikroTik berubah hijau pada siklus polling berikutnya. Skenario ini menunjukkan bahwa Zabbix dapat mengisolasi service-level fault pada satu perangkat tanpa memengaruhi konfigurasi host lainnya.

SNMP availability pada host Miktorik berubah merah, sedangkan Forgitae di VyOS tetap hijau. Hasil ini membuktikan bahwa Zabbix mampu mendeteksi gangguan pada satu perangkat tanpa memengaruhi status perangkat lain, sehingga monitoring tidak hanya menampilkan data secara statis.

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : https://doi.org/10.58602/itsecs.v4i2.390
Page 30-41



Gambar 6. Deteksi Gangguan SNMP Miktorik pada Zabbix

3.6. Ringkasan Hasil Pengujian

Sebanyak 15 skenario diuji dari validasi awal hingga workflow end-to-end. Seluruh skenario memperoleh status PASS. Fungsi jaringan mencakup DHCP, konektivitas gateway, akses internet, resolusi DNS, transit routing, dan policy NAT. Workflow end-to-end berakhir dengan failed=0 dan unreachable=0. Pada tahap verification, actual state dapat dibaca kembali dan dibandingkan dengan desired state. Temuan ini menunjukkan bahwa prototipe memenuhi tujuan fungsional dalam lingkungan laboratorium yang ditetapkan.

Tabel 5. Ringkasan Seluruh Skenario Pengujian

No	Skenario	Hasil Aktual	Status
1	Validasi koneksi multi-vendor	Tiga perangkat reachable dan identitas sesuai	PASS
2	Backup konfigurasi	RSC, CONF, dan snapshot JSON tersedia	PASS
3	Deployment MikroTik	VLAN, DHCP, DNS, route, SNMP sesuai	PASS
4	Deployment VyOS	Interface, static route, SNMP sesuai	PASS
5	Deployment FortiGate	Route, object, NAT policy, SNMP sesuai	PASS
6	Idempotency deployment	Eksekusi ulang changed=0	PASS
7	Verification	Assertion berhasil tanpa perubahan	PASS
8	Controlled failure	Deviasi alias FortiGate terdeteksi	PASS
9	Controlled failure	Alias awal dipulihkan dan diverifikasi	PASS
10	Registrasi Zabbix	Tiga host dan template tersedia	PASS
11	Idempotency Zabbix	Eksekusi kedua changed=0; tanpa duplikasi	PASS
12	Polling SNMP	Tiga indikator availability hijau	PASS
13	Fault detection	SNMP MikroTik menjadi unavailable	PASS
14	Recovery	Availability kembali hijau	PASS
15	Workflow end-to-end	failed=0 dan unreachable=0	PASS

3.7. Analisa Perbandingan dan Pembahasan

Hasil penelitian memperlihatkan bahwa kontribusi utama prototipe berada pada integrasi lifecycle, bukan pada pengenalan teknologi otomasi baru. Penelitian terdahulu telah menunjukkan keberhasilan Ansible untuk routing, VLAN, EIGRP, virtual network, SDN, dan backup [3], [4], [11]–[13], sementara Zabbix berbasis SNMP telah digunakan untuk monitoring terpusat [9]. Prototipe ZTBD menggabungkan fungsi-fungsi tersebut

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

dengan validasi, backup, verification, idempotency, rollback, dan registrasi monitoring dalam satu alur. Temuan ini sejalan dengan arah zero-touch management yang menekankan pengurangan intervensi manual [1].

Perbedaan karakteristik vendor juga memberikan temuan penting. Pada MikroTik dan VyOS, idempotency memerlukan pembacaan current state dan kondisi when karena penggunaan command dapat bersifat imperatif dan format konfigurasi dapat menghasilkan false positive changed. FortiGate lebih sesuai dengan modul resource-specific melalui HTTPAPI, tetapi membutuhkan ketelitian pada token, selector, dan object identifier. Dengan demikian, hasil penelitian mendukung pandangan bahwa otomasi multi-vendor tidak cukup diselesaikan dengan satu playbook generik; abstraksi diperlukan pada level workflow, sedangkan implementasi task tetap mengikuti model data dan mekanisme koneksi setiap platform [7], [8], [14].

Dari sisi monitoring, integrasi API membuat provisioning host menjadi bagian dari workflow otomasi, sedangkan polling SNMP menyediakan observability pasca-deployment. Hasil fault injection menunjukkan bahwa gangguan layanan pada satu perangkat dapat dideteksi tanpa mengubah konfigurasi host lain. Hal ini memperkuat manfaat monitoring terpusat yang dilaporkan penelitian sebelumnya [9], [15]. Namun, SNMPv2c masih memiliki keterbatasan keamanan karena community string tidak memberikan perlindungan autentikasi dan kerahasiaan seperti SNMPv3 [16].

Dari sisi keamanan dan operasional, penggunaan Ansible Vault, no_log, permission backup 0600, direktori 0700, dan pembatasan MariaDB pada localhost memberikan kontrol dasar terhadap credential dan artefak. Namun, prototipe belum mencakup audit trail enterprise, approval workflow, role-based access, centralized logging, physical-device scale testing, atau pengukuran waktu manual dibandingkan otomatis. Karena itu, hasil PASS tidak boleh ditafsirkan sebagai bukti kesiapan produksi penuh. Hasil penelitian lebih tepat dipahami sebagai validasi fungsional terhadap pola workflow ZTBD pada skala laboratorium, yang dapat menjadi dasar untuk pengujian lebih lanjut pada perangkat fisik, jumlah node lebih besar, latency, packet loss, dynamic routing, VPN, high availability, SNMPv3, CI/CD, dan notifikasi operasional.

4. KESIMPULAN

Penelitian ini menunjukkan bahwa Ansible dapat digunakan sebagai pusat otomasi konfigurasi dan integrasi monitoring perangkat jaringan multi-vendor pada lingkungan enterprise yang disimulasikan. Prototipe ZTBD berhasil mengelola MikroTik RouterOS, VyOS, dan FortiGate melalui satu control node, sedangkan Zabbix memantau ketiga perangkat menggunakan SNMPv2c. Workflow terintegrasi mencakup validasi, backup, deployment, verification, idempotency, registrasi monitoring, controlled failure, rollback, dan recovery. Seluruh 15 skenario pengujian memperoleh status PASS. Eksekusi ulang deployment menghasilkan changed=0 pada ketiga vendor, sedangkan workflow end-to-end berakhir dengan failed=0 dan unreachable=0. Controlled failure pada alias FortiGate berhasil mengaktifkan rescue, memulihkan state sebelumnya, dan memverifikasi kondisi akhir. Integrasi Zabbix juga berhasil mendaftarkan tiga host tanpa duplikasi serta mendeteksi dan memulihkan gangguan SNMP MikroTik.

Keterbatasan utama penelitian adalah penggunaan lingkungan virtual dengan tiga perangkat dan cakupan konfigurasi yang masih dasar. Pengembangan selanjutnya disarankan menggunakan perangkat fisik dan jumlah node lebih besar, mengukur waktu eksekusi serta penggunaan sumber daya, dan menguji latency atau packet loss. Cakupan layanan dapat diperluas ke dynamic routing, VPN, high availability, multiple VLAN, QoS, dan segmentasi firewall. Untuk kesiapan produksi, SNMPv2c sebaiknya ditingkatkan ke

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

SNMPv3, akses Zabbix menggunakan HTTPS, credential menerapkan least privilege dan rotasi berkala, serta workflow diintegrasikan dengan version control, CI/CD, approval, centralized logging, audit trail, dashboard, dan notifikasi operasional

REFERENCES

[1] E. Coronado *et al.*, "Zero Touch Management: A Survey of Network Automation Solutions for 5G and 6G Networks," *IEEE Commun. Surv. \& Tutorials*, pp. 2535–2578, 2022, doi: 10.13039/501100011033.

[2] M. Lyu, H. Habibi Gharakheili, and V. Sivaraman, "A Survey on Enterprise Network Security: Asset Behavioral Monitoring and Distributed Attack Detection," *IEEE Access*, vol. 12, pp. 89363–89383, 2024, doi: 10.1109/ACCESS.2024.3419068.

[3] E. N. Fadhila, E. R. Gumelar, H. R. Pratama, and G. M. Suranegara, "Otomasi Konfigurasi Routing pada Router menggunakan Ansible," *TELNECT E-Journal Univ. Pendidik. Indones.*, 2021.

[4] M. F. Mohd Fuzi, K. Abdullah, I. H. Abd Halim, and R. Ruslan, "Network Automation using Ansible for EIGRP Network," *J. Comput. Res. Innov.*, vol. 6, no. 4, pp. 59–69, 2021, doi: 10.24191/jcrinn.v6i4.237.

[5] N. M. A. Yalestia Chandrawaty and I. P. Hariyadi, "Implementasi Ansible Playbook Untuk Mengotomatisasi Manajemen Konfigurasi VLAN Berbasis VTP Dan Layanan DHCP," *J. Bumigora Inf. Technol.*, vol. 3, no. 2, pp. 107–122, Dec. 2021, doi: 10.30812/bite.v3i2.1577.

[6] M. Fahmi, M. Maisyaroh, K. Ishak, S. Faizah, and I. Fadhillah, "Otomatisasi Jaringan Menggunakan Script Python Untuk Penyediaan Konfigurasi Internet Dan Manajemen Mikrotik," *Bina Insa. ICT J.*, vol. 8, no. 1, pp. 53–62, 2021.

[7] E. Mulyana and G. Fakih, "Network Automation with a Single Source of Truth in a Heterogeneous Environment," *Int. J. Electr. Eng. Informatics*, vol. 14, no. 1, pp. 92–100, 2022, doi: 10.15676/ijeei.2022.14.1.6.

[8] S. Nugroho and B. Pujiarto, "Network Automation pada Beberapa Perangkat Router Menggunakan Pemrograman Python," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 9, no. 1, pp. 79–86, 2022, doi: 10.25126/jtiik.202293947.

[9] A. Pradana, I. R. Widiasari, R. Efendi, and T. Informatika, "Implementasi Sistem Monitoring Jaringan Menggunakan Zabbix Berbasis SNMP," *AITI J. Teknol. Inf.*, vol. 19, pp. 248–262, 2022.

[10] W. R. A. Putra, A. R. A. Nurwa, D. F. Priambodo, and M. Hasbi, "Infrastructure as Code for Security Automation and Network Infrastructure Monitoring," *MATRIK J. Manajemen, Tek. Inform. dan Rekayasa Komput.*, vol. 22, no. 1, pp. 201–214, 2022, doi: 10.30812/matrik.v22i1.2471.

[11] K. Marzuki, M. I. Kholid, I. P. Hariyadi, and L. Z. A. Mardedi, "Automation of Open VSwitch-Based Virtual Network Configuration Using Ansible on Proxmox Virtual Environment," *Int. J. Electron. Commun. Syst.*, vol. 3, no. 1, pp. 11–20, 2023, doi: 10.24042/ijecs.v3i1.16524.

[12] K. Marzuki, H. Husain, and Z. Apriliana, "Automation of Software Defined Network (SDN) Configuration Management Based on Proxmox with Ansible," *Int. J. Electron. Commun. Syst.*, vol. 4, no. 2, pp. 87–97, 2024, doi: 10.24042/ijecs.v4i2.23821.

[13] A. Zulfikar and Y. Akbar, "Otomasi Backup Konfigurasi Settingan Router Mikrotik Menggunakan Ansible dengan Metode Network DevOps," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 5, no. 1, pp. 57–66, 2024, doi: 10.57152/malcom.v5i1.1591.

[14] Fandi Ali Mustika, Muhammad Rifqi, Yuwan Jumaryadi, Febryo Ponco Sulistyo, Indah Ramadhani, and Eko Prasetyo Pratomo, "Implementation of Proxy at XYZ Inc. Study: Experiment on Network Performance Optimization," *PIKSEL Penelit. Ilmu Komput.*

Fandi Ali Mustika: *Penulis Korespondensi

Journal of Information Technology, Software Engineering and Computer Science
Website : https://ejournal.techcart-press.com/index.php/itsecs
E-ISSN 2962-0635
P-ISSN 2962-0538
Volume 4, Number 2, April 2026
DOI : **https://doi.org/10.58602/itsecs.v4i2.390**
Page 30-41

Sist. Embed. Log., vol. 13, no. 2, pp. 55–62, Sep. 2025, doi: 10.33558/piksel.v13i2.11535.

[15] A. B. Imoukhuede, T. R. Sheltami, A. H. Mahmoud, and A. Y. Barnawi, "Optimization of network device hardening in a multivendor environment," *Sci. Rep.*, vol. 15, no. 1, 2025, doi: 10.1038/s41598-025-97894-4.

[16] A. Wahab, F. A. Mustika, R. B. Bahaweres, D. Setiawan, and M. Alaydrus, "Energy efficiency and loss of transmission data on Wireless Sensor Network with obstacle," in *2016 10th International Conference on Telecommunication Systems Services and Applications (TSSA)*, IEEE, Oct. 2016, pp. 1–6. doi: 10.1109/TSSA.2016.7871084.

[17] T. Ariyadi, M. Fikri, and H. Yudiastuti, "Penerapan Monitoring Jaringan Dengan Zabbix Pada PT. PLN (Persero) UIP Bagian Sumbagsel," *J. Ilm. Inform. Univ. Puter. Batam*, 2024.

[18] A. Kodar, F. A. Mustika, and A. Wahab, "Utilizing Genetic Algorithm for Optimization of Maritime Surveillance Sensor Network Deployment," in *2019 IEEE 6th International Conference on Engineering Technologies and Applied Sciences (ICETAS)*, IEEE, Dec. 2019, pp. 1–6. doi: 10.1109/ICETAS48360.2019.9117220.

[19] Cisco Systems, Inc., "Configure SNMPv2c/v3 on Catalyst 9000 Series Switches," 2025.

[20] I. Kumara *et al.*, "The Do's and Don'ts of Infrastructure Code: a Systematic Grey Literature Review," 2021.

[21] F. Rezabek *et al.*, "EnGINE: Flexible Research Infrastructure for Reliable and Scalable Time Sensitive Networks," *J. Netw. Syst. Manag.*, vol. 30, no. 4, 2022, doi: 10.1007/s10922-022-09686-0.

[22] D. Susanto, R. Ferdiana, and S. Sulistyo, "Implementasi Laboratorium Komputer Virtual Berbasis Cloud-Kelas Pemrograman Berorientasi Obyek," *J. Nas. Tek. Elektro dan Teknol. Inf.*, vol. 11, no. 1, 2022.

[23] Zabbix, "Zabbix Documentation 7.0," 2024. [Online]. Available: https://www.zabbix.com/documentation/7.0/en/manual

Fandi Ali Mustika: *Penulis Korespondensi