

**ANALISIS PERFORMA NETWORK AUTOMATION ENGINE
PADA CLUSTER SERVER MENGGUNAKAN METODE
TOPSIS (TECHNIQUE FOR ORDER OF PREFERENCE BY
SIMILARITY TO IDEAL SOLUTION)**

SKRIPSI

**Diajukan Untuk Menglengkapi Salah Satu Syarat
Memperoleh Gelar Sarjana Komputer**



OLEH :

ABDUL QUDDUS PAHML SZ

09011282126068

JURUSAN SISTEM KOMPUTER

FAKULTAS ILMU KOMPUTER

UNIVERSITAS SRIWIJAYA

2025

AUTHENTICATION PAGE

SKRIPSI

***PERFORMANCE ANALYSIS OF NETWORK AUTOMATION
ENGINES ON CLUSTER SERVERS USING THE TOPSIS
METHOD (TECHNIQUE FOR ORDER OF PREFERENCE BY
SIMILARITY TO IDEAL SOLUTION)***

As one of the requirements for completing the Bachelor's
Degree Program in Computer Systems.

By:

**ABDUL QUDDUS PAHMI. SZ
09011282126068**

**Supervisor 1 : Dr. Ir. Ahmad Hervanto., S.Kom., M.T
NIP. 198701222015041002**

**Approved by,
Head of Computer System Department**



**Dr. Ir. Sukemi, M.T
196612032006041001**

HALAMAN PENGESAHAN

SKRIPSI

ANALISIS PERFORMA NETWORK AUTOMATION ENGINE
PADA CLUSTER SERVER MENGGUNAKAN METODE
TOPSIS (TECHNIQUE FOR ORDER OF PREFERENCE BY
SIMILARITY TO IDEAL SOLUTION)

Sebagai salah satu syarat untuk penyelesaian studi di

Program Studi S1 Sistem Komputer

By:

ABDUL QUDDUS PAHMI. SZ

09011282126068

Pembimbing 1 : Dr. Ir. Ahmad Heryanto., S.Kom., M.T
NIP. 198701222015041002

Mengetahui
Ketua Jurusan Sistem Komputer



Dr. Ir. Sukemi, M.T
196612032006041001

HALAMAN PERSETUJUAN

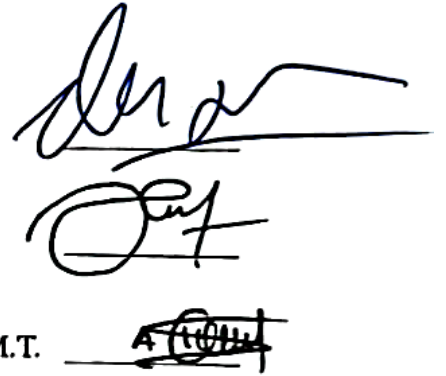
Telah diuji pada:

Hari : Jum'at

Tanggal : 11 Juli 2025

Tim Penguji:

1. Ketua : Prof. Deris Stiawan, M.T., Ph.D.
2. Penguji : Ahmad Fali Oklilas, M.T.
3. Pembimbing I : Dr. Ir. Ahmad Heryanto., S.Kom., M.T.



Mengetahui, 12/8/25

Ketua Jurusan Sistem Komputer



Dr. Ir. Sukemi, M.T.

NIP. 196612032006041001

HALAMAN PERNYATAAN

Yang bertanda tangan :

Nama : Abdul Quddus Pahmi. SZ

NIM : 09011282126068

Judul : Analisis Performa *Network Automation Engine* Pada *Cluster Server* Menggunakan Metode TOPSIS (*Technique For Order of Preference By Similarity to Ideal Solution*)

Hasil Pengecekan Software iThenticate/Turnitin : 1%

Menyatakan bahwa laporan tugas akhir saya merupakan hasil karya saya sendiri dan bukan hasil penjiplakan/plagiat. Apabila ditemukan unsur penjiplakan/plagiat dalam laporan tugas akhir ini, maka saya bersedia menerima sanksi akademik dari Universitas Sriwijaya.

Demikian pernyataan ini saya buat dalam keadaan sadar dan tidak dipaksakan.



Palembang, Agustus 2025



Abdul Quddus Pahmi. SZ

NIM. 09011282126018

***Performance Analysis of Network Automation Engines on Cluster Servers
Using the TOPSIS Method (Technique for Order of Preference by Similarity to
Ideal Solution)***

Abdul Quddus Pahmi. SZ (09011282126068)

Dept. Of Computer System, Faculty of Computer Science, Sriwijaya University

Email : abdulpahmi55@gmail.com

ABSTRACT

The advancement of network technology drives the need for configuration automation in server infrastructure, especially within complex cluster server environments. This study aims to analyze and compare the performance of three *network automation* engines, namely *Ansible*, *Puppet*, and *Chef*, in automating server services (*Apache*, *MySQL*, *Exim*, and *FileZilla*) on two virtualization platforms: *Proxmox* and *VMware ESXi*. The study was conducted through several stages, including literature review, virtual infrastructure design, testing of 24 scenarios before and after optimization, and result evaluation using five parameters: CPU usage, RAM usage, *data transfer speed*, *amount of data sent*, and *configuration time*. The *TOPSIS* method was used to determine the tool with the most efficient performance. The results show that the *PR_PU_FTP* scenario (*Puppet* on *FTP* service in *Proxmox*) achieved the highest score of 0.9076, indicating the most optimal performance. In addition, the *EX_CH_WEB* scenario (*Chef* on *Apache* service in *VMware ESXi*) showed consistent and stable results after being tested 10 times, with an average CPU usage of 0.22%, RAM usage of 10.2%, configuration time of 44.7 seconds, and data transfer speed of 1138.1 kb/s. These findings provide a strong foundation for selecting the most suitable combination of tool, service, and platform for efficient and reliable server system automation.

Keyword : *Network Automation, Cluster Server, Ansible, Puppet, Chef, TOPSIS*

**Analisis Performa *Network Automation Engine* Pada *Cluster Server*
Menggunakan Metode TOPSIS (*Technique For Order of Preference By
Similarity to Ideal Solution*)**

Abdul Quddus Pahmi. SZ (09011282126068)

Dept. Of Computer System, Faculty of Computer Science, Sriwijaya University

Email : abdulpahmi55@gmail.com

ABSTRACT

Kemajuan teknologi jaringan mendorong kebutuhan akan otomatisasi konfigurasi pada infrastruktur server, terutama dalam lingkungan kluster *server* yang kompleks. Penelitian ini bertujuan untuk menganalisis dan membandingkan performa tiga alat otomatisasi jaringan, yaitu *Ansible*, *Puppet*, dan *Chef*, dalam mengotomatisasi layanan server (*Apache*, *MySQL*, *Exim*, dan *FileZilla*) pada dua platform virtualisasi, yaitu Proxmox dan VMware ESXi. Penelitian dilakukan melalui beberapa tahapan, termasuk studi literatur, perancangan infrastruktur virtual, pengujian 24 skenario sebelum dan sesudah optimalisasi, serta evaluasi hasil menggunakan lima parameter: penggunaan CPU, penggunaan RAM, kecepatan transfer data, jumlah data yang dikirim, dan waktu konfigurasi. Metode *TOPSIS* digunakan untuk menentukan alat dengan performa paling efisien. Hasil penelitian menunjukkan bahwa skenario *PR_PU_FTP* (*Puppet* pada layanan *FTP* di *Proxmox*) memperoleh skor tertinggi sebesar 0,9076, menandakan kinerja paling optimal. Selain itu, skenario *EX_CH_WEB* (*Chef* pada layanan *Apache* di *VMware ESXi*) menunjukkan hasil yang konsisten dan stabil setelah diuji ulang sebanyak 10 kali, dengan rata-rata penggunaan CPU 0,22%, RAM 10,2%, waktu konfigurasi 44,7 detik, dan kecepatan transfer data 1138,1 kb/s. Temuan ini memberikan landasan kuat dalam memilih kombinasi alat, layanan, dan platform yang paling tepat untuk otomatisasi sistem server yang efisien dan andal.

Keyword : Otomatisasi Jaringan, Kluster *Server*, *Ansible*, *Puppet*, *Chef*, *TOPSIS*

KATA PENGANTAR

Puji syukur penulis panjatkan ke hadirat Allah SWT, Tuhan semesta alam, karena atas limpahan rahmat, hidayah, dan karunia-Nya, penulis dapat menyelesaikan skripsi yang berjudul “**Analisis Performa *Network Automation Engine* pada *Cluster Server* Menggunakan Metode TOPSIS (*Technique for Order of Preference by Similarity to Ideal Solution*)**” dengan baik. Skripsi ini disusun sebagai salah satu syarat untuk menyelesaikan studi pada Program Studi Sistem Komputer, Fakultas Ilmu Komputer, Universitas Sriwijaya. Penulisan dan penyusunan skripsi ini tentu bukanlah hal yang mudah. Banyak tantangan teknis maupun non-teknis yang harus dihadapi, terutama dalam tahap implementasi dan pengujian performa sistem otomatisasi jaringan. Namun, semua itu menjadi pengalaman berharga yang memperkaya proses belajar dan pendewasaan diri penulis.

Penulis menyadari bahwa terselesaikannya skripsi ini tidak lepas dari bantuan, dukungan, doa, dan bimbingan dari banyak pihak. Oleh karena itu, dengan segala kerendahan hati, penulis ingin menyampaikan ucapan terima kasih kepada :

1. Allah SWT, atas segala berkah dan nikmat yang tak ternilai.
2. Diri sendiri, karena telah bertahan, belajar, dan terus mencoba walau berkali-kali gagal.
3. Moto hidup "Selalu belajar dari kesalahan", yang menjadi pengingat dalam setiap proses.
4. Orang tua tercinta – ibu, ayah, serta saudara dan saudariku – atas doa, dukungan moral dan spiritual yang tak pernah henti.
5. Bapak Dr. Ir. H. Sukemi, M.T., selaku Ketua Jurusan Sistem Komputer Universitas Sriwijaya, yang telah memberikan fasilitas dan arahan dalam menyelesaikan studi ini.
6. Bapak Abdurahman, S.Kom., M.Han, selaku dosen pembimbing akademik yang selalu memberi motivasi dan panduan selama masa studi.

7. Bapak Dr. Ir. Ahmad Heryanto, S.Kom., M.T, selaku dosen pembimbing skripsi yang telah dengan sabar membimbing, mengarahkan, dan memberikan kritik konstruktif sejak awal hingga akhir penyusunan skripsi ini.
8. Teman-teman seperjuangan SKB 2021, yang selalu menjadi tempat berbagi semangat dan tawa di tengah tekanan.
9. Rekan-rekan dari grub "Crowd Control", atas bantuan teknis dan support dalam pengembangan server Comnets Indralaya.
10. Rekan-rekan dari grub "Bawak Laptop Dak", yang selalu menemani proses debugging dengan gelak tawa.
11. Rekan-rekan dari grub "Botanisme", tempat recharge semangat saat burnout melanda.
12. Sahabat selamanya dari persatuan batalion "Duo Trio" yang beranggotakan saya sendiri, Bagas Firmansyah, dan Dhani Medianto Saputra.
13. Serta semua teman-teman dan pihak lain yang tidak dapat disebutkan satu per satu, namun telah memberikan kontribusi besar dalam proses ini.

Palembang, Agustus 2025

Penulis,



Abdul Quddus Pahmi. SZ

NIM. 0901182126018

DAFTAR ISI

AUTHENTICATION PAGE.....	i
HALAMAN PENGESAHAN	ii
HALAMAN PERSETUJUAN	Error! Bookmark not defined.
HALAMAN PERNYATAAN	iii
KATA PENGANTAR	vii
DAFTAR ISI	ix
DAFTAR GAMBAR	xi
DAFTAR TABEL	xiii
DAFTAR LAMPIRAN.....	xiv
BAB I PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Rumusan Masalah	4
1.3 Batasan Masalah	4
1.4 Tujuan	4
1.5 Manfaat.....	5
1.6 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA	7
2.1 Penelitian Terdahulu	7
2.2 <i>Network Automation</i>	13
2.3 <i>Infrastructure as a Code (IaC)</i>	14
2.4.1 Ansible	15
2.4.2 Puppet.....	18
2.4.3 Chef.....	19
2.4 Parameter Evaluasi	21
2.4.1 Parameter Pemakaian CPU	22
2.4.2 Parameter Data yang Terkirim (Bytes)	23
2.4.3 Parameter Pemakaian RAM.....	23
2.4.4 Parameter Bandwidth	24
2.4.5 Parameter Waktu Konfigurasi	24
2.5 Metode TOPSIS (<i>Technique For Order of Preference by Similarity to Ideal Solution</i>).....	25
BAB III METODOLOGI PENELITIAN.....	28
3.1 Kerangka Kerja Penelitian	28
3.2 Perancangan Topologi <i>Network Automation</i>	31
3.3 Perancangan Skenario Pengujian	33
3.4 Implementasi Topologi dan Skenario <i>Network Automation</i>	35
3.4.1 Lingkungan Implementasi	35
3.4.2 Kebutuhan Perangkat pada Topologi.....	38
3.4.3 Konfigurasi <i>Network Automation</i> pada Skenario	40
3.5 Model Pengumpulan Data	46
3.5.1 Persiapan.....	46
3.5.2 Pengumpulan Data Konfigurasi Layanan Server Skenario Ansible	50
3.5.3 Pengumpulan Data Konfigurasi Layanan Server Skenario Puppet	53
3.5.4 Pengumpulan Data Konfigurasi Layanan Server Skenario Chef	57

3.6	Analisis Data	60
3.6.1	Matriks Evaluasi	61
3.6.2	Normalisasi	63
3.6.3	Normalisasi Berbobot	66
3.6.4	Menentukan Solusi Ideal Positif dan Negatif	68
3.6.5	Jarak Solusi Ideal	73
3.6.6	Nilai Preferensi dan Perangkingan	74
BAB IV	HASIL DAN ANALISIS	77
4.1	Analisis Performa 24 Skenario Default	77
4.1.1	Parameter Pemakaian CPU (%)	78
4.1.2	Parameter Data yang Terkirim	83
4.1.3	Parameter Pemakaian RAM	87
4.1.4	Parameter Bandwidth (Kb/s)	90
4.1.5	Parameter Waktu konfigurasi	93
4.2	Hasil Evaluasi Menggunakan Metode TOPSIS	96
4.2.1	Penerapan Metode TOPSIS pada Hasil Pengujian	97
4.2.2	Perhitungan Nilai Preferensi dan Peringkat Akhir	97
4.2.3	Validasi Skor Preferensi dan Peringkat Akhir	99
4.3	Identifikasi Dampak Optimalisasi	100
4.3.1	Perbandingan Singkat Default vs Optimalisasi (24 skenario)	100
4.3.2	Rincian Parameter dan Komponen yang Dioptimalisasi pada Setiap Alat	109
4.3.3	Identifikasi Skenario Optimalisasi Paling Berdampak	111
4.4	Hasil Pengujian Ulang pada Skenario Optimalisasi	113
4.4.1	Distribusi dan Konsistensi Data	113
4.4.2	Implikasi Terhadap Stabilitas Performansi	115
BAB V	KESIMPULAN DAN SARAN	116
5.1	Kesimpulan	116
5.2	Saran	117
DAFTAR PUSTAKA		119
LAMPIRAN		127

DAFTAR GAMBAR

Gambar 2. 1 Struktur Playbook Ansible	17
Gambar 2. 2 Struktur Manifest Puppet	19
Gambar 2. 3 Struktur Cookbook Chef.....	21
Gambar 3. 1 Kerangka Kerja Penelitian	29
Gambar 3. 2 Gambaran Model Penelitian	30
Gambar 3. 3 Rancangan Sederhana Topologi.....	32
Gambar 3. 4 Antarmuka Platform Proxmox	36
Gambar 3. 5 Antarmuka Platform Vmware ESXI	36
Gambar 3. 6 Struktur File Ansible	41
Gambar 3. 7 Struktur File Puppet.....	43
Gambar 3. 8 Struktur File Chef.....	44
Gambar 3. 9 Gambaran Hasil Konfigurasi Apache Webserver.....	47
Gambar 3. 10 Gambaran Hasil Konfigurasi MySQL Database Server	47
Gambar 3. 11 Gambaran Hasil Konfigurasi Exim Mail Server	48
Gambar 3. 12 Gambaran Hasil Konfigurasi FileZilla FTP Server	48
Gambar 3. 13 Gambaran Pengambilan Data melalui Grafana.....	49
Gambar 3. 14 Traffic data Tcpdump.....	49
Gambar 4. 1 Perbandingan Pemakaian CPU Basic (%) - Apache	78
Gambar 4. 2 Perbandingan Pemakaian CPU Basic (%) - MySQL	79
Gambar 4. 3 Perbandingan Pemakaian CPU Basic (%) - Exim.....	79
Gambar 4. 4 Perbandingan Pemakaian CPU Basic (%) - FileZilla.....	80
Gambar 4. 5 Perbandingan Pemakaian CPU Busy (%) - Apache	81
Gambar 4. 6 Perbandingan Pemakaian CPU Busy (%) - MySQL	81
Gambar 4. 7 Perbandingan Pemakaian CPU Busy (%) - Exim	82
Gambar 4. 8 Perbandingan Data yang Tekirim (%) - FileZilla.....	83
Gambar 4. 9 Perbandingan Transferred Data (MB) - Apache	84
Gambar 4. 10 Perbandingan Transferred Data (MB) - MySQL	85
Gambar 4. 11 Perbandingan Transferred Data (MB) - Exim.....	86
Gambar 4. 12 Perbandingan Transferred Data (MB) – FileZilla	87
Gambar 4. 13 Perbandingan Ram Usage (%) - Apache	88
Gambar 4. 14 Perbandingan Ram Usage (%) - MySQL	88
Gambar 4. 15 Perbandingan Ram Usage (%) - Exim.....	89
Gambar 4. 16 Perbandingan Ram Usage (%) - FileZilla.....	90
Gambar 4. 17 Perbandingan Bandwidth Per Minute (Kb/s) - Apache	91
Gambar 4. 18 Perbandingan Bandwidth Per Minute (Kb/s) - MySQL	91
Gambar 4. 19 Perbandingan Bandwidth Per Minute (Kb/s) - Exim	92
Gambar 4. 20 Perbandingan Bandwidth Per Minute (Kb/s) - FileZilla	93
Gambar 4. 21 Perbandingan Configuration Time (s) - Apache	94
Gambar 4. 22 Perbandingan Bandwidth Per Minute (Kb/s) - MySQL	94
Gambar 4. 23 Perbandingan Bandwidth Per Minute (Kb/s) - Exim	95
Gambar 4. 24 Perbandingan Bandwidth Per Minute (Kb/s) - FileZilla	96
Gambar 4. 25 Parameter Data yang Terkirim (MB) Sebelum Optimalisasi	101
Gambar 4. 26 Parameter Penggunaan RAM Sebelum Optimalisasi	102

Gambar 4. 27	Parameter CPU Usage (%) Sebelum Optimalisasi.....	102
Gambar 4. 28	Parameter CPU Busy Data (%) Sebelum Optimalisasi	103
Gambar 4. 29	Parameter Bandwidth (Kb/s) Sebelum Optimalisasi.....	103
Gambar 4. 30	Parameter Waktu Konfigurasi (s) Sebelum Optimalisasi	104
Gambar 4. 31	Parameter Data yang Terkirim Setelah Optimalisasi.....	106
Gambar 4. 32	Parameter Penggunaan RAM (%) Setelah Optimalisasi	107
Gambar 4. 33	Parameter Bandwidth (Kb/s) Setelah Optimalisasi	107
Gambar 4. 34	Parameter CPU Usage (%) Setelah Optimalisasi.....	108
Gambar 4. 35	Parameter Waktu Konfigurasi (s) Setelah Optimalisasi	108
Gambar 4. 36	Parameter CPU Busy Data (%) Setelah Optimalisasi	109

DAFTAR TABEL

Tabel 2. 1 Penelitian terdahulu yang dijadikan sebagai acuan.....	7
Tabel 3. 1 Skenario Perpaduan Alat Otomatisasi Server	33
Tabel 3. 2 Spesifikasi Fisik Server Proxmox dan VMware ESXi.....	37
Tabel 3. 3 Spesifikasi Mesin Virtual (VM) Uji Coba.....	37
Tabel 3. 4 Kebutuhan Perangkat Fisik.....	38
Tabel 3. 5 Layanan Apache Webserver Skenario Ansible	50
Tabel 3. 6 Layanan MySQL Database Server Skenario Ansible	51
Tabel 3. 7 Layanan Exim Mail Server Skenario Ansible	52
Tabel 3. 8 Layanan FileZilla FTP Server Skenario Ansible	53
Tabel 3. 9 Layanan Apache Webserver Skenario Puppet.....	54
Tabel 3. 10 Layanan MySQL Database Server Skenario Puppet.....	55
Tabel 3. 11 Layanan Exim Mail Server Skenario Puppet.....	55
Tabel 3. 12 Layanan FileZilla FTP Server Skenario Puppet	56
Tabel 3. 13 Layanan Apache Webserver Skenario Chef.....	57
Tabel 3. 14 Layanan MySQL Database Server Skenario Chef.....	58
Tabel 3. 15 Layanan Exim Mail Server Skenario Chef.....	59
Tabel 3. 16 Layanan FileZilla FTP Server Skenario Chef.....	60
Tabel 3. 17 Data Penilaian Alternatif	61
Tabel 3. 18 Bobot Parameter.....	63
Tabel 3. 19 Matriks Evaluasi kuadrat	64
Tabel 3. 20 Matriks Normalisasi	65
Tabel 3. 21 Matriks Normalisasi Berbobot.....	67
Tabel 3. 22 Matriks Solusi Ideal	70
Tabel 3. 23 Matriks Solusi Ideal Positif	71
Tabel 3. 24 Matriks Solusi Ideal Negatif.....	72
Tabel 3. 25 Total Jarak Positif dan Negatif.....	73
Tabel 3. 26 Nilai Preferensi dan Peringkat	75
Tabel 4. 1 Nilai Preferensi dan Peringkat Akhir.....	98
Tabel 4. 2 Nilai Preferensi dan Peringkat Akhir.....	99
Tabel 4. 3 Nilai Preferensi dan Peringkat Akhir	111
Tabel 4. 4 Hasil Pengujian Ulang Skenario EX_CH_WEB (10x)	113

DAFTAR LAMPIRAN

Lampiran 1. Konfigurasi Ansible - Web, Database, Mail, dan FTP Server	128
Lampiran 2. Konfigurasi Optimalisasi Layanan Server Ansible	133
Lampiran 3. Konfigurasi Puppet - Web, Database, Mail, dan FTP Server.....	139
Lampiran 4. Konfigurasi Optimalisasi Layanan Server Puppet	146
Lampiran 5. Konfigurasi Chef - Web, Database, Mail, dan FTP Server	151
Lampiran 6. Konfigurasi Optimalisasi Layanan Server Chef	159
Lampiran 7. Cek Plagiarisme	164
Lampiran 8. Form Revisi Penguji	165
Lampiran 9. Form Revisi Pembimbing I.....	166

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kemajuan teknologi di bidang server berkembang dengan pesat, memungkinkan seorang administrator untuk mengelola banyak server sekaligus. Namun, ini juga menghadirkan tantangan tambahan dalam pengelolaan server yang menjadi semakin kompleks [1] [2]. Pada awalnya, konfigurasi sistem dikelola secara manual oleh administrator, membutuhkan upaya yang besar dan pencatatan sistematis tentang status sistem. Seiring dengan meningkatnya kompleksitas kebutuhan, otomatisasi dalam pengelolaan konfigurasi server kini menjadi sangat penting, dan hal ini dapat dicapai melalui berbagai alat manajemen konfigurasi yang tersedia [3].

Network Automation merupakan konsep yang berkaitan erat dengan Infrastructure as Code (IaC), yang bertujuan untuk menghindari kesalahan yang dilakukan pengelolaan sistem secara manual [4]. Penerapan *Network Automation* sebagai alat pendukung dalam pengelolaan jaringan dapat meningkatkan produktivitas dan menurunkan jumlah terjadinya kesalahan karena semua konfigurasi ditulis dan di sajikan dalam bentuk skrip [5].

Pengelolaan dan pengkonfigurasian yang berkelanjutan mencakup sejumlah elemen penting dalam pengelolaan sistem dan infrastruktur sistem. Termasuk dalam hal ini adalah penerapan otomatisasi konfigurasi, manajemen konfigurasi, dan pendekatan modern yang dikenal sebagai *Infrastructure as Code* (IaC). Dalam upaya mencapai pengelolaan dan pengkonfigurasian yang berkelanjutan, organisasi sering mengandalkan berbagai alat atau vendor otomatisasi yang berfokus dalam pengelolaan dan konfigurasi lanjutan pada infrastruktur TI [6].

Ada berbagai macam alat manajemen konfigurasi yang dapat ditemukan di pasaran, seperti Chef, Puppet, Salt, dan Ansible, yang masing-masing dirancang

untuk tujuan yang berbeda dan menawarkan fitur yang beragam [3]. Beberapa diantaranya adalah Ansible, Ansible merupakan alat otomatisasi TI [7]. Ansible dirancang untuk melakukan operasi di banyak server dan merupakan alat otomatisasi yang sangat baik untuk penerapan manajemen konfigurasi. Biasanya disebutkan bersamaan dengan alat otomatisasi lain seperti Chef, Puppet, dan Salt [6].

Penelitian oleh Mohammed Daffa Elradi [8] mengungkapkan bahwa ansible sangatlah kuat dan penuh fitur yang menjadikannya alat yang andal untuk digunakan mengotomatiskan berbagai tugas secara tepat waktu yang merupakan faktor penting dalam mengelola infrastruktur TI. Selain itu, penelitian oleh Katsuno dan Hitomi [9] mengeksplorasi *Infrastructure as Service* (IaaS) dalam pengelolaan jaringan, mereka menunjukkan bahwa Chef membuat pengaturan server dan aplikasi menjadi lebih cepat dan mudah dengan mengatasi dependensi antara mereka. Alat otomatisasi dapat membantu menyelesaikan berbagai masalah yang telah dijelaskan sebelumnya dengan membuat pengelolaan dan pengaturan server lebih mudah bagi staf pemeliharaan, khususnya untuk kluster server [10].

Sebagaimana ditunjukkan dalam beberapa penelitian terdahulu, otomasi dalam konfigurasi jaringan dan server telah dianggap sebagai solusi utama untuk meningkatkan efisiensi operasional serta mengurangi kesalahan manusia, terutama pada infrastruktur yang membutuhkan manajemen yang kompleks dan waktu implementasi yang cepat. Penelitian [11] membahas perbandingan kinerja antara Ansible dan alat berbasis Python dengan fokus pada analisis kualitatif mengenai kemudahan penggunaan dan fleksibilitas masing-masing alat. Namun, penelitian ini tidak memberikan pendekatan metodologi yang terstruktur atau validasi berbasis data, sehingga kesimpulan yang diberikan tidak didukung dengan analisis performa yang komprehensif. Sementara itu, penelitian [2] lebih menekankan pada pengurangan waktu pengaturan infrastruktur server dengan menggunakan Ansible sebagai kerangka kerja otomasi. Penelitian ini menggarisbawahi manfaat kecepatan yang dihasilkan, tetapi tidak menjelaskan metode yang digunakan untuk

peningkatan waktu secara mendetail, seperti metrik pengujian, skenario uji, atau validasi statistik dari hasil yang diperoleh. Demikian pula, penelitian [3] berfokus pada implementasi Ansible untuk konfigurasi server otomatis dengan menggambarkan tahapan teknis penerapan, tetapi tidak menyediakan evaluasi performa berbasis data atau metode analisis untuk mengukur efisiensi atau efektivitas solusi tersebut. Ketiga penelitian tersebut, meskipun memberikan kontribusi dalam deskripsi implementasi, memiliki kelemahan mendasar dalam tahap analisis data karena tidak adanya validasi melalui metode yang terstruktur, seperti pengujian asumsi statistik atau perhitungan signifikan yang dapat mendukung hasil penelitian. Untuk mengatasi kekurangan ini, penelitian ini akan menggunakan Metode TOPSIS sebagai pendekatan evaluasi yang berlandaskan pada anggapan bahwa pilihan yang diambil harus memiliki jarak terpendek dari Solusi Ideal Positif (PIS) dan jarak terjauh dari Solusi Ideal Negatif (NIS). Peringkat akhir ditetapkan melalui indeks kedekatan [12].

TOPSIS didasarkan pada konsep bahwa alternatif ideal memiliki tingkat terbaik untuk semua atribut yang dipertimbangkan, sedangkan alternatif negatif-ideal adalah yang memiliki nilai terburuk untuk semua atribut. Solusi dari TOPSIS didefinisikan sebagai alternatif yang secara bersamaan paling jauh dari negatif-ideal dan paling dekat dengan alternatif ideal [13]. Pada penelitian Ümran Sengül dkk [14], mereka melakukan percobaan kinerja metode TOPSIS dan hasil evaluasi menunjukkan bahwa metode tersebut sangat baik dalam meningkatkan kemampuan perankingan yang memungkinkan penilaian alternatif terhadap berbagai kriteria. Dengan menggunakan metode ini, analisis akan dilakukan secara terstruktur dan valid, memastikan hasil yang lebih akurat dan dapat diandalkan dalam menentukan keunggulan pendekatan otomatisasi jaringan dan server.

Pada penelitian ini, penulis akan berfokus pada analisis kinerja masing-masing alat otomatisasi infrastruktur, yaitu Ansible, Chef, Puppet, pada Proxmox dan VMware ESXi pada kluster server ketika diterapkan dalam konfigurasi layanan menggunakan metode analisis TOPSIS. Dengan memahami pentingnya pengkonfigurasiannya yang efisien dalam infrastruktur TI, penelitian ini akan

mengeksplorasi bagaimana ketiga alat ini berkinerja dalam mengotomatisasi konfigurasi di Virtual Environment yang berbeda.

1.2 Rumusan Masalah

Rumusan masalah penelitian tugas akhir ini adalah sebagai berikut:

1. Bagaimana performa alat network automation engine seperti Ansible, Chef, dan Puppet dalam mengotomatisasi konfigurasi pada cluster server dengan lingkungan virtual yang berbeda-beda?
2. Bagaimana metode TOPSIS dapat digunakan untuk menilai dan menentukan alat network automation engine yang paling optimal berdasarkan parameter performa seperti CPU, RAM, bandwidth, dan waktu konfigurasi?

1.3 Batasan Masalah

Beberapa batasan masalah dari penelitian tugas akhir ini adalah sebagai berikut:

1. Parameter pengujian penelitian ini adalah total pemakaian CPU, total data yang dikirim (Bytes), total RAM yang digunakan, rata-rata bandwidth, dan total waktu konfigurasi.
2. Penelitian ini menganalisis penggunaan sumber daya server dan waktu yang diperlukan untuk konfigurasi.
3. Metode yang digunakan adalah TOPSIS.

1.4 Tujuan

Setelah merumuskan masalah latar belakang, tujuan penelitian ini adalah sebagai berikut:

1. Menganalisis dan membandingkan performa Ansible, Chef, dan Puppet

dalam mengotomatisasi konfigurasi pada cluster server dengan virtual environment yang berbeda-beda.

2. Mengukur efisiensi penggunaan sumber daya server oleh masing-masing alat otomasi berdasarkan parameter performa.
3. Menggunakan metode TOPSIS untuk menentukan alat otomasi terbaik dalam konteks konfigurasi otomatis berbasis *network automation*.

1.5 Manfaat

Beberapa manfaat dari penelitian tugas akhir ini adalah sebagai berikut:

1. Memberikan perbandingan kinerja alat otomasi (Ansible, Chef, Puppet) dalam manajemen konfigurasi server.
2. Mengidentifikasi efisiensi dan kinerja masing-masing alat dalam penggunaan sumber daya server dan waktu konfigurasi.
3. Memberikan rekomendasi alat otomasi terbaik berdasarkan hasil analisis kinerja dan efisiensi pemakaian sumber daya menggunakan metode TOPSIS.

1.6 Sistematika Penulisan

Berikut sistematika penulisan penelitian tugas akhir ini :

BAB I PENDAHULUAN

Pada bab ini membahas secara garis besar mengenai latar belakang, tujuan penelitian, manfaat penelitian, rumusan masalah, metode Penulisan dan batasan masalah.

BAB II TINJAUAN PUSTAKA

Pada bab ini membahas teori-teori yang relevan pada penelitian ini, mencakup konsep *Network Automation* dan Infrastructure as Code (IaC), alat otomatisasi (Ansible, Chef, Puppet), dan virtual environment

yang digunakan (Proxmox dan VMware ESXi), serta metode yang digunakan dalam penelitian.

BAB III METODOLOGI PENELITIAN

Pada bab ini menguraikan tentang pendekatan yang digunakan dalam penelitian ini, termasuk pengumpulan data, parameter yang diukur (seperti penggunaan CPU, RAM, bandwidth, waktu konfigurasi dan total data yang ditransfer), serta metode analisis yang diterapkan, yaitu TOPSIS.

BAB IV HASIL PENGUJIAN DAN ANALISA

Pada bab ini mencakup uraian hasil pengujian yang dilakukan pada alat-alat otomatisasi konfigurasi (Ansible, Chef, Puppet), serta analisis performa masing-masing alat berdasarkan parameter yang telah ditentukan.

BAB V KESIMPULAN

Merupakan bagian akhir dari penelitian yang berisi kesimpulan dari hasil penelitian, serta saran untuk pengembangan lebih lanjut terkait penggunaan alat otomatisasi dalam pengelolaan server. Juga diberikan rekomendasi alat otomatisasi terbaik berdasarkan analisis yang telah dilakukan.

DAFTAR PUSTAKA

- [1] A. C. Kheirabadi and D. Groulx, "Cooling of server electronics: A design review of existing technology," *Appl. Therm. Eng.*, vol. 105, no. March, pp. 622–638, 2016, doi: 10.1016/j.applthermaleng.2016.03.056.
- [2] B. Santoso and M. W. Sari, "IMPROVEMENT OF SETUP TIME ON SERVER INFRASTRUCTURE AUTOMATION USING ANSIBLE FRAMEWORK," *J. Eng. Sci. Technol.*, vol. 17, no. 5, pp. 3660–3671, 2022.
- [3] L. S, "Automation of Server Configuration Using Ansible," *Int. J. Res. Appl. Sci. Eng. Technol.*, vol. 10, no. 6, pp. 4109–4113, 2022, doi: 10.22214/ijraset.2022.44840.
- [4] K. Morris, *Infrastructure as code: managing servers in the cloud*, First Edit. Sebastopol: O'Reilly Media, Inc., 2016. [Online]. Available: <https://www.oreilly.com/library/view/infrastructure-as-code/9781491924334/>
- [5] M. F. Islami, P. Musa, and M. Lamsani, "Implementation of Network Automation using Ansible to Configure Routing Protocol in Cisco and Mikrotik Router with Raspberry PI," *J. Ilm. Komputasi*, vol. 19, no. 2, pp. 127–134, 2020, doi: 10.32409/jikstik.19.2.80.
- [6] P. T. P, C. S, and D. M. R, "Devops Methods for Automation of Server Management using Ansible," *Int. J. Adv. Sci. Innov.*, vol. 1, no. 2, May 2021, [Online]. Available: www.ijasi.org
- [7] G. L and S. K N, "Mathematical tools and methods for analysis of SDN: A comprehensive survey," in *Mathematical Tools and Methods for Analysis of SDN: A Comprehensive Survey*, IEEE, 2016, pp. 718–724. doi: <https://doi.org/10.1109/IC3I.2016.7918055>.
- [8] M. Daffalla Elradi, "Electrical and Computer Engineering Studies Ansible: A Reliable Tool for Automation," *Electr. Comput. Eng. Stud.*, vol. 2, no. 1, pp. 1–10, Aug. 2023.
- [9] Y. Katsuno and H. Takahashi, "An automated parallel approach for rapid deployment of composite application servers," in *Proceedings - 2015 IEEE International Conference on Cloud Engineering, IC2E 2015*, Institute of Electrical and Electronics Engineers Inc., 2015, pp. 126–134. doi: 10.1109/IC2E.2015.16.
- [10] W. Sha, M. Wu, and Y. Qiao, "Research on Automated Operation and Maintenance Methods for Power Management Networks Based on Ansible," *2023 5th Int. Conf. Artif. Intell. Comput. Appl. ICAICA 2023*, pp. 591–595, 2023, doi: 10.1109/ICAICA58456.2023.10405593.
- [11] N. Aldeen Younes, "Network Automation: A Comparative Analysis of

- Ansible and a Custom Python-Based Tool,” Mälardalen University, 2024. [Online]. Available: <https://urn.kb.se/resolve?urn=urn:nbn:se:mdh:diva-67588>
- [12] M. S. García-Cascales and M. T. Lamata, “On rank reversal and TOPSIS method,” *Math. Comput. Model.*, vol. 56, no. 5–6, pp. 123–132, 2012, doi: 10.1016/j.mcm.2011.12.022.
 - [13] T. C. Chu and Y. C. Lin, “A fuzzy TOPSIS method for robot selection,” *Int. J. Adv. Manuf. Technol.*, vol. 21, no. 4, pp. 284–290, 2003, doi: 10.1007/s001700300033.
 - [14] Ü. Şengül, M. Eren, S. Eslamian Shiraz, V. Gezder, and A. B. Sengül, “Fuzzy TOPSIS method for ranking renewable energy supply systems in Turkey,” *Renew. Energy*, vol. 75, pp. 617–625, 2015, doi: 10.1016/j.renene.2014.10.045.
 - [15] A. M. Mazin, R. A. Rahman, M. Kassim, and A. R. Mahmud, “Performance analysis on network automation interaction with network devices using python,” in *ISCAIE 2021 - IEEE 11th Symposium on Computer Applications and Industrial Electronics*, Institute of Electrical and Electronics Engineers Inc., Apr. 2021, pp. 360–366. doi: 10.1109/ISCAIE51753.2021.9431823.
 - [16] E. Luchian, C. Filip, A. B. Rus, I. A. Ivanciu, and V. Dobrota, “Automation of the infrastructure and services for an openstack deployment using chef tool,” *Netw. Educ. Res. RoEduNet Int. Conf. 15th Ed. RoEduNet 2016 - Proc.*, 2016, doi: 10.1109/RoEduNet.2016.7753200.
 - [17] D. Badalyan and O. Borisenko, “Ansible execution control in Python and Golang for cloud orchestration,” *SoftwareX*, vol. 19, Jul. 2022, doi: 10.1016/j.softx.2022.101126.
 - [18] S. D. Palma, D. Di Nucci, F. Palomba, and D. A. Tamburri, “Within-Project Defect Prediction of Infrastructure-as-Code Using Product and Process Metrics,” *IEEE Trans. Softw. Eng.*, vol. 48, no. 6, pp. 2086–2104, 2022, doi: 10.1109/TSE.2021.3051492.
 - [19] M. M. Hassan and A. Rahman, “As Code Testing: Characterizing Test Quality in Open Source Ansible Development,” *Proc. - 2022 IEEE 15th Int. Conf. Softw. Testing, Verif. Validation, ICST 2022*, pp. 208–219, 2022, doi: 10.1109/ICST53961.2022.00031.
 - [20] I. Pinto, *Network Automation Made Easy*. Cisco Press, 2021. [Online]. Available: https://books.google.co.id/books?hl=id&lr=&id=ptfPEAAAQBAJ&oi=fnd&pg=PT23&dq=network+automation+iac&ots=kEvu5RdlOS&sig=OxlE2nvzL-kezurF37Yl-lwhTmc&redir_esc=y#v=onepage&q=network+automation&f=false

- [21] A. F. Rochim, M. A. Aziz, and A. Fauzi, "Design Log Management System of Computer Network Devices Infrastructures Based on ELK Stack," *ICECOS 2019 - 3rd Int. Conf. Electr. Eng. Comput. Sci. Proceeding*, pp. 338–342, 2019, doi: 10.1109/ICECOS47637.2019.8984494.
- [22] A. P. Paramitha, A. F. Rochim, and A. Fauzi, "Design and Implementation Network Administrators Account Management System Based on Authentication, Authorization, and Accounting Based on TACACS and LDAP," *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 803, no. 1, 2020, doi: 10.1088/1757-899X/803/1/012040.
- [23] A. F. Rochim, A. Rafi, A. Fauzi, and K. T. Martono, "As-RaD System as a Design Model of the Network Automation Configuration System Based on the REST-API and Django Framework," *Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control*, vol. 4, pp. 291–298, 2020, doi: 10.22219/kinetik.v5i4.1093.
- [24] A. (2017). P. N. A. L. the power of P. and A. to optimize your network. I. P. P. A. Ratan, "Practical Network Automation: Leverage the power of Python and Ansible to optimize your network," *Packt Publishing*. Packt Publishing Ltd, 2017. [Online]. Available: https://books.google.co.id/books?id=jEFPDwAAQBAJ&dq=network+automation+google+books&lr=&hl=id&source=gb_s_navlinks_s
- [25] S. Nelson-smith, *Test-Driven Infrastructure with Chef: Bring Behavior-Driven Development to Infrastructure as Code*. "O'Reilly Media, Inc.," 2013. [Online]. Available: https://books.google.co.id/books?id=fT1PAQAAQBAJ&dq=Test-Driven+Infrastructure+with+Chef+bring+behavior-driven+development+to+infrastructure+as+code&lr=&hl=id&source=gb_s_navlinks_s
- [26] A. Rahman and C. Parnin, "Detecting and Characterizing Propagation of Security Weaknesses in Puppet-Based Infrastructure Management," *IEEE Trans. Softw. Eng.*, vol. 49, no. 6, pp. 3536–3553, 2023, doi: 10.1109/TSE.2023.3265962.
- [27] J. Sandobalin, E. Insfran, and S. Abrahao, "On the effectiveness of tools to support infrastructure as code: Model-driven versus code-centric," *IEEE Access*, vol. 8, pp. 17734–17761, 2020, doi: 10.1109/ACCESS.2020.2966597.
- [28] M. Guerriero, M. Garriga, D. A. Tamburri, and F. Palomba, "Adoption, Support, and Challenges of Infrastructure-as-Code: Insights from Industry," *Proc. - 2019 IEEE Int. Conf. Softw. Maint. Evol. ICSME 2019*, pp. 580–589, 2019, doi: 10.1109/ICSME.2019.00092.
- [29] Y. Brikman, *Terraform Up and Running*. "O'Reilly Media, Inc.," 2022. [Online]. Available:

https://books.google.co.id/books?id=dlaKEAAQBAJ&dq=Terraform:+Up+and+Running&lr=&hl=id&source=gbs_navlinks_s

- [30] M. M. Hasan, F. A. Bhuiyan, and A. Rahman, "Testing practices for infrastructure as code," *LANGETI 2020 - Proc. 1st ACM SIGSOFT Int. Work. Lang. Tools Next-Generation Testing, Co-located with ESEC/FSE 2020*, pp. 7–12, 2020, doi: 10.1145/3416504.3424334.
- [31] A. Rahman, R. Mahdavi-Hezaveh, and L. Williams, "A systematic mapping study of infrastructure as code research," *Inf. Softw. Technol.*, vol. 108, no. i, pp. 65–77, 2019, doi: 10.1016/j.infsof.2018.12.004.
- [32] Andrew Crump and others of Foodcritic, "About Foodcritic." Accessed: Dec. 14, 2024. [Online]. Available: <http://www.foodcritic.io/>
- [33] S. Dalla Palma, D. Di Nucci, F. Palomba, and D. A. Tamburri, "Toward a catalog of software quality metrics for infrastructure code," *J. Syst. Softw.*, vol. 170, p. 110726, 2020, doi: 10.1016/j.jss.2020.110726.
- [34] M. Artac, T. Borovssak, E. Di Nitto, M. Guerriero, and D. A. Tamburri, "DevOps: Introducing infrastructure-as-code," *Proc. - 2017 IEEE/ACM 39th Int. Conf. Softw. Eng. Companion, ICSE-C 2017*, no. May, pp. 497–498, 2017, doi: 10.1109/ICSE-C.2017.162.
- [35] M. Hüttermann, *Infrastructure as Code*. Springer, 2012. doi: 10.1109/MS.2022.3212034.
- [36] D. Soldani, B. Barani, R. Tafazolli, A. Manzalini, and C. L. I, "Software defined 5G networks for anything as a service [Guest Editorial]," *IEEE Commun. Mag.*, vol. 53, no. 9, pp. 72–73, 2015, doi: 10.1109/MCOM.2015.7263348.
- [37] L. Hochstein and R. Moser, *Ansible: Up and Running: Automating Configuration Management and Deployment the Easy Way*, 2nd Edition., vol. 2. Gravenstein Highway North, Sebastopol, CA 95472: O'Reilly Media, 2017.
- [38] G. Drosos and E. T. H. Zurich, "When Your Infrastructure is a Buggy Program : Understanding Faults in Infrastructure as Code Ecosystems," *Proc. ACM Program. Lang.*, vol. 8, no. OOPSLA2, 2024, doi: 10.1145/3689799.
- [39] I. Nagaraju, "Automating Infrastructure Management : The Power of Ansible for DevOps Efficiency," *Int. J. Comput. Tech.*, vol. 7, no. 2, pp. 1–14, 2020, [Online]. Available: <http://www.ijctjournal.org/archives/ijct-v7i2p18.pdf>
- [40] S. Mysari and V. Bejgam, "Continuous Integration and Continuous Deployment Pipeline Automation Using Jenkins Ansible," *Int. Conf. Emerg. Trends Inf. Technol. Eng. ic-ETITE 2020*, no. March, 2020, doi: 10.1109/ic-ETITE47903.2020.239.

- [41] Ansible Community, “Ansible community documentation.” Accessed: Dec. 19, 2024. [Online]. Available: <https://docs.ansible.com/>
- [42] Inc. StackExchange, “2023 Developer Survey.” Accessed: Dec. 19, 2024. [Online]. Available: <https://survey.stackoverflow.co/2023/#section-most-popular-technologies-other-tools>
- [43] Red Hat Ansible Team, “Ansible PyPI Package.” Accessed: Dec. 19, 2024. [Online]. Available: <https://pypi.org/project/ansible/>
- [44] C. Flynn, “Ansible PyPI Package Stats.” Accessed: Dec. 19, 2024. [Online]. Available: <https://pypistats.org/packages/ansible>
- [45] Red Hat Ansible Team, “Ansible GitHub Organization.” Accessed: Dec. 19, 2024. [Online]. Available: <https://github.com/ansible>
- [46] P. Sahoo, S. Pujar, G. Nalawade, R. Gebhardt, L. Mandel, and L. Buratti, “Ansible Lightspeed: A Code Generation Service for IT Automation,” *Proc. ACM Conf.*, vol. 1, pp. 2148–2158, 2024, doi: 10.1145/3691620.3695277.
- [47] E. Van Der Bent, J. Hage, J. Visser, and G. Gousios, “How good is your puppet? An empirically defined and validated quality model for puppet,” *25th IEEE Int. Conf. Softw. Anal. Evol. Reengineering, SANER 2018 - Proc.*, vol. 2018-March, pp. 164–174, 2018, doi: 10.1109/SANER.2018.8330206.
- [48] J. Rhett, *Learning Puppet 4: A Guide to Configuration Management and Automation*, 4th ed. “O’Reilly Media, Inc.,” 2016. [Online]. Available: https://books.google.co.id/books?id=W7zRCwAAQBAJ&dq=puppet+infrastructure+as+code&lr=&hl=id&source=gbp_navlinks_s
- [49] J. Loope, *Managing Infrastructure with Puppet*. “O’Reilly Media, Inc.,” 2011. [Online]. Available: <http://books.google.com/books?hl=en&lr=&id=TCfZ7d6skT4C&oi=fnd&pg=PR5&dq=Think+Stats&ots=LxYK9ntvRZ&sig=UchH878Uf04hPLGPmOwnNHBGmdk>
- [50] E. Waud, *Mastering Chef Provisioning*. Packt Publishing Ltd, 2016. [Online]. Available: https://books.google.co.id/books?id=m_9vDQAAQBAJ&dq=chef+infrastructure+as+code&lr=&hl=id&source=gbp_navlinks_s
- [51] J. Cowie, *Customizing Chef: Getting the Most Out of Your Infrastructure Automation*. “O’Reilly Media, Inc.,” 2014. [Online]. Available: https://books.google.co.id/books?id=8fldBAAAQBAJ&dq=chef+infrastructure+as+code&lr=&hl=id&source=gbp_navlinks_s
- [52] S. Pandya and R. G. Thakurta, *Introduction to Infrastructure as Code with Chef*. Springer, 2022. doi: 10.1007/978-1-4842-8689-0.

- [53] E. Hancer, "Fuzzy kernel feature selection with multi-objective differential evolution algorithm," *Conn. Sci.*, vol. 31, no. 4, pp. 323–341, 2019, doi: 10.1080/09540091.2019.1639624.
- [54] R. Kamimura and H. Takeuchi, "Improving collective interpretation by extended potentiality assimilation for multi-layered neural networks," *Conn. Sci.*, vol. 32, no. 2, pp. 174–203, 2020, doi: 10.1080/09540091.2019.1674245.
- [55] K. Jacksi and R. M. Zeebaree, "Effects of Processes Forcing on CPU and Total Execution-Time Using Multiprocessor Shared Memory System," *Int. J. Comput. Eng. Res. Trends*, vol. 2, no. 4, pp. 275–279, 2015, [Online]. Available: https://www.ijcert.org/issue_des.php?id=312
- [56] Z. N. Rashid, H. Sharif, S. Rafeeq, and M. Z. Sulaimani, "Client/Servers Clustering Effects on CPU Execution-Time, CPU Usage and CPU Idle Depending on Activities of Parallel-Processing-Technique Operations "," *Int. J. Sci. Technol. Res.*, vol. 7, no. 8, 2018, [Online]. Available: www.ijstr.org
- [57] M. Lubis, A. Widjajarto, A. Fiqhiyah, R. Ismah Ramadhani, and A. Ridho Lubis, "Configuration and Performance Management of Ansible to Deploy CMS Platform and Development of Odoo Server Automation Engine," *Artic. Internetworking Indones. J.*, vol. 13, no. 1, 2021, [Online]. Available: <https://www.researchgate.net/publication/366095698>
- [58] N. M. N. Mathivanan, N. A. Nor, and R. M. Janor, "Improving classification accuracy using clustering technique," *Bull. Electr. Eng. Informatics*, vol. 7, no. 3, pp. 465–470, 2018, doi: 10.11591/eei.v7i3.1272.
- [59] R. Fauzi, M. Hariadi, S. M. S. Nugroho, and M. Lubis, "Defense behavior of real time strategy games: Comparison between HFSM and FSM," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 13, no. 2, pp. 634–642, 2019, doi: 10.11591/ijeecs.v13.i2.pp634-642.
- [60] A. Widjajarto, D. W. Jacob, and M. Lubis, "Live migration using checkpoint and restore in userspace (Criu): Usage analysis of network, memory and CPU," *Bull. Electr. Eng. Informatics*, vol. 10, no. 2, pp. 837–847, 2021, doi: 10.11591/eei.v10i2.2742.
- [61] N. Tolia, M. Kaminsky, D. G. Andersen, and S. Patil, "An architecture for internet data transfer," *3rd Symp. Networked Syst. Des. Implementation, NSDI 2006*, no. May, 2006.
- [62] E. E. Hassan, T. K. A. Rahman, Z. Zakaria, N. Bahaman, and M. H. Jifri, "Maximum loadability enhancement with a hybrid optimization method," *Bull. Electr. Eng. Informatics*, vol. 7, no. 3, pp. 323–330, 2018, doi: 10.11591/eei.v7i3.1168.
- [63] J. T, "Ram versus CAM," *IEEE Potentials*, vol. 16, no. 2 Suppl, pp. 26–28,

1997, doi: 10.1109/45.580445.

- [64] C. Wang, S. Zhang, Y. Chen, Z. Qian, J. Wu, and M. Xiao, "Joint Configuration Adaptation and Bandwidth Allocation for Edge-based Real-time Video Analytics," *Proc. - IEEE INFOCOM*, vol. 2020-July, pp. 257–266, 2020, doi: 10.1109/INFOCOM41043.2020.9155524.
- [65] X. Yang *et al.*, "Fast and light bandwidth testing for internet users," *Proc. 18th USENIX Symp. Networked Syst. Des. Implementation, NSDI 2021*, pp. 1011–1026, 2021.
- [66] H. Deng, C. Peng, A. Fida, J. Meng, and Y. Charlie Hu, "Mobility support in cellular networks: A measurement study on its configurations and implications," *Proc. ACM SIGCOMM Internet Meas. Conf. IMC*, pp. 147–160, 2018, doi: 10.1145/3278532.3278546.
- [67] Y. Li *et al.*, "Icellular: Device-customized cellular network access on commodity smartphones," *Proc. 13th USENIX Symp. Networked Syst. Des. Implementation, NSDI 2016*, pp. 643–656, 2016.
- [68] M. Raquibul Hasan, M. Sifuddin Ansary, and N. Biz, "Cloud Infrastructure Automation Through IaC (Infrastructure as Code)," *Int. J. Comput.*, vol. 46, no. 1, pp. 34–40, 2023.
- [69] D. Sokolowski, "Infrastructure as code for dynamic deployments," *ESEC/FSE 2022 - Proc. 30th ACM Jt. Meet. Eur. Softw. Eng. Conf. Symp. Found. Softw. Eng.*, pp. 1775–1779, 2022, doi: 10.1145/3540250.3558912.
- [70] S.-J. Chen and C.-L. Hwang, *Fuzzy Multiple Attribute Decision Making Methods*, no. XI. Springer, 1992. doi: 10.1007/978-3-642-46768-4_5.
- [71] J. H. Park, I. Y. Park, Y. C. Kwun, and X. Tan, "Extension of the TOPSIS method for decision making problems under interval-valued intuitionistic fuzzy environment," *Appl. Math. Model.*, vol. 35, no. 5, pp. 2544–2556, 2011, doi: 10.1016/j.apm.2010.11.025.
- [72] T. Y. Chen and C. Y. Tsao, "The interval-valued fuzzy TOPSIS method and experimental analysis," *Fuzzy Sets Syst.*, vol. 159, no. 11, pp. 1410–1428, 2008, doi: 10.1016/j.fss.2007.11.004.
- [73] J. Puttonen and G. Fekete, "Interface selection for multihomed mobile hosts," *IEEE Int. Symp. Pers. Indoor Mob. Radio Commun. PIMRC*, 2006, doi: 10.1109/PIMRC.2006.254042.
- [74] H. Jia, Z. Zhang, P. Cheng, H. H. Chen, and S. Li, "Study on network selection for next-generation heterogeneous wireless networks," *IEEE Int. Symp. Pers. Indoor Mob. Radio Commun. PIMRC*, 2006, doi: 10.1109/PIMRC.2006.253971.
- [75] G. Gódor and G. Détári, "Novel network selection algorithm for various wireless network interfaces," *2007 16th IST Mob. Wirel. Commun. Summit*,

2007, doi: 10.1109/ISTMWC.2007.4299136.

- [76] Q. Song and A. Jamalipour, "A network selection mechanism for next generation networks," *IEEE Int. Conf. Commun.*, vol. 2, no. C, pp. 1418–1422, 2005, doi: 10.1109/icc.2005.1494578.
- [77] G.-H. Tzeng and J.-J. Huang, "Multiple Attribute Decision Making: Methods and Applications." CRC Press, 2011.
- [78] K. Yoon and C.-L. Hwang, *Multiple Attribute Decision Making: An Introduction*. SAGE Publications, 1995. doi: 10.4135/9781412985161.
- [79] S. Chakraborty, "A Simulation Based Comparative Study of Normalization Procedures in Multiattribute Decision Making," *6th WSEAS Int. Conf. Artif. Intell. Knowl. Eng. Data Bases*, vol. Proceeding, pp. 102–109, 2007.