

Implementasi Sistem *Monitoring* Server Menggunakan Prometheus dan Grafana dengan Notifikasi Email

Umi Nasiyatun Nauha¹, Alif Subardono^{1*}

¹Departemen Teknik Elektro dan Informatika, Sekolah Vokasi, Universitas Gadjah Mada;
umi.nasiyatun1903@mail.ugm.ac.id

*Korespondensi: alif@ugm.ac.id;

Abstract – *The ever-evolving technological era makes companies increasingly dependent on IT infrastructure, which makes server monitoring an important aspect of maintaining operational stability. The process of manually monitoring 74 servers is ineffective, making it difficult for administrators to detect anomalies in real-time. PT HMMI has used Zabbix to monitor all network devices, however, server performance decreases when a server monitoring dashboard is added. This research will implement a server monitoring system using Prometheus and Grafana with email notifications. Prometheus collects metrics data such as CPU, memory, storage, and throughput centrally, while Grafana provides interactive and easy-to-understand data visualization. This system replaces inefficient manual methods with the purpose of making it easier for administrators to monitor server resources in real-time. The test results show that the Grafana monitoring system successfully displays real-time data and sends email notifications with a 100% success rate. In addition, the accuracy of the test data shows CPU usage reaches 95.20%, memory 99.71%, disk 99.71%, and throughput 91.42%. Server performance testing recorded the highest CPU usage of 2.11%, the maximum memory of 22.48%, the largest disk space of 20.85%, and the highest throughput reaching 203.56 Kbps when monitoring six servers for six hours. The system proved effective in detecting anomalies and helped maintain the stability of the company's services..*

Keywords – *Monitoring servers, Prometheus, Grafana*

Intisari – Era teknologi yang terus berkembang membuat perusahaan semakin bergantung pada infrastruktur IT yang menjadikan *monitoring* server sebagai aspek penting untuk menjaga stabilitas operasional. Proses memantau 74 server secara manual tidak efektif sehingga menyulitkan administrator mendeteksi anomali secara *real-time*. PT HMMI telah menggunakan Zabbix untuk *monitoring* seluruh perangkat jaringan namun, performa server menurun ketika ditambahkan *dashboard monitoring* server. Pada penelitian ini akan dilakukan implementasi sistem *monitoring* server menggunakan Prometheus dan Grafana dengan notifikasi email. Prometheus mengumpulkan data metrik seperti CPU, memori, penyimpanan, dan *throughput* secara terpusat, sementara Grafana menyediakan visualisasi data yang interaktif dan mudah dipahami. Sistem ini menggantikan metode manual yang kurang efisien dengan tujuan guna mempermudah administrator dalam memantau *resource* server secara *real-time*. Hasil pengujian menunjukkan bahwa sistem *monitoring* Grafana berhasil menampilkan data secara *real-time* dan mengirimkan notifikasi email dengan tingkat keberhasilan 100%. Selain itu, akurasi data pengujian menunjukkan penggunaan CPU mencapai 95,20%, memori 99,71%, *disk* 99,71%, dan *throughput* 91,42%. Pengujian performa server mencatat penggunaan CPU tertinggi sebesar 2,11%, memori maksimum 22,48%, ruang *disk* terbesar 20,85%, dan *throughput* tertinggi mencapai 203,56 Kbps saat memantau enam server selama enam jam. Sistem ini terbukti efektif dalam mendeteksi anomali dan membantu menjaga stabilitas layanan perusahaan.

Kata kunci – *Monitoring server, Prometheus, Grafana*

I. PENDAHULUAN

Dalam era transformasi digital yang terus berkembang, teknologi informasi (TI) untuk menjalankan operasional perusahaan [1]. Ketergantungan pada infrastruktur TI, khususnya server, semakin meningkat seiring dengan kompleksitas sistem dan kebutuhan akan pengolahan data yang cepat dan efisien. Server tidak hanya berfungsi sebagai pusat penyimpanan dan pengolahan data, tetapi juga sebagai penyedia layanan yang memastikan kelancaran berbagai proses bisnis. Oleh karena itu, menjaga kinerja dan stabilitas server menjadi hal yang sangat krusial untuk meminimalkan risiko *downtime* yang dapat mengganggu operasional perusahaan.

Monitoring server merupakan salah satu upaya strategis untuk memastikan bahwa server beroperasi secara optimal [2]. Proses ini mencakup pengumpulan data secara *real-time* mengenai penggunaan sumber daya seperti CPU, memori, penyimpanan, dan *throughput* [3]. Namun, pemantauan

manual sering kali tidak efektif karena sulitnya mendeteksi anomali secara cepat, terutama pada lingkungan dengan jumlah server yang besar. Untuk itu, diperlukan solusi *monitoring* otomatis yang mampu memberikan visibilitas menyeluruh terhadap kinerja server sekaligus menyediakan notifikasi dini ketika terjadi masalah.

PT HMMI menghadapi tantangan serupa dalam mengelola 74 server produksinya. Penggunaan alat *monitoring* sebelumnya, seperti Zabbix, terbukti kurang optimal dalam hal performa dan visualisasi data. Selain itu, ketergantungan pada metode manual menyebabkan keterlambatan dalam mendeteksi anomali yang berdampak pada gangguan operasional. Sebagai perusahaan non-TI, PT HMMI membutuhkan solusi *monitoring* yang tidak hanya andal tetapi juga mudah diintegrasikan dengan sistem komunikasi internal perusahaan.

Penelitian ini bertujuan untuk mengimplementasikan sistem *monitoring* berbasis Prometheus dan Grafana sebagai

solusi alternatif yang lebih efektif dan efisien. Prometheus memungkinkan pengumpulan data secara terpusat dengan kemampuan analisis *real-time*, sementara Grafana menyediakan visualisasi data yang intuitif untuk mempermudah interpretasi informasi oleh administrator server. Selain itu, integrasi notifikasi melalui email dirancang untuk memberikan peringatan dini kepada administrator sehingga tindakan korektif dapat segera diambil.

Melalui penerapan sistem ini, diharapkan PT HMMI dapat meningkatkan efisiensi operasionalnya dengan meminimalkan risiko *downtime* serta memastikan infrastruktur TI berfungsi secara optimal dalam mendukung aktivitas bisnis perusahaan.

II. DASAR TEORI

A. Tinjauan Pustaka

Penelitian [2] mengembangkan sistem *monitoring* server menggunakan Prometheus dan Grafana dengan notifikasi Telegram. Sistem ini dirancang untuk memantau kondisi server secara *real-time* pada Ubuntu Server 18.04, fokus pada CPU, memori, dan layanan Apache serta MySQL. Hasilnya menunjukkan peningkatan respons terhadap anomali server dan ketersediaan jaringan, dengan notifikasi yang berhasil dikirim ke administrator saat *resource* server melewati batas yang ditentukan.

Penelitian [4] mengimplementasikan sistem *monitoring resource* server berbasis Prometheus dan Grafana dengan notifikasi Telegram. Sistem ini diterapkan pada Ubuntu Server 18.04 dan CentOS Server 7, menggunakan Node Exporter untuk mengekspos metrik. Grafana memvisualisasikan data dari Prometheus, sementara Telegram mengirimkan notifikasi saat terjadi anomali. Pengujian menunjukkan tingkat keberhasilan 100% dalam fungsionalitas dan pengiriman notifikasi.

Penelitian [5] merancang dan menerapkan solusi pemantauan menggunakan platform Zabbix untuk perangkat server di PT. Rizki Tujuh Belas Kelola. Metodologi mencakup analisis kebutuhan, konfigurasi perangkat, dan pengujian sistem. Hasil menunjukkan peningkatan kinerja server, deteksi masalah proaktif, dan notifikasi melalui email saat terjadi anomali. Implementasi ini meningkatkan visibilitas kinerja jaringan dan mempercepat respons terhadap masalah.

Penelitian [6] mengimplementasikan sistem *monitoring* server berbasis Android menggunakan OpenTelemetry dan Prometheus. Aplikasi dikembangkan dengan Kotlin, menggunakan REST API Node.js dan Express.js. Sistem berhasil melakukan *monitoring real-time* dengan notifikasi

melalui *Firebase Cloud Messaging*. Pengujian mencakup fungsionalitas, pengaruh terhadap penggunaan CPU dan memori server, serta tampilan antarmuka. Hasilnya menunjukkan efektivitas dalam meningkatkan visibilitas kinerja server dan efisiensi administrator.

Penelitian [3] merancang sistem *monitoring* server di PT. XYZ Media Indonesia menggunakan Prometheus, Grafana, dan Node Exporter. Sistem ini memantau parameter seperti CPU, *storage*, memori, dan jaringan secara *real-time*, dengan *alert* otomatis melalui Telegram. Pengumpulan data log server meningkatkan efisiensi *monitoring* yang sebelumnya manual. Hasil pengujian menunjukkan efektivitas dalam memberikan informasi aktivitas server, memungkinkan respons cepat terhadap masalah dan memudahkan *monitoring* jarak jauh.

B. Dasar Teori

1. Monitoring Server

Monitoring server adalah proses pemantauan kinerja server secara *real-time* untuk memastikan operasional yang optimal. Proses ini mencakup pengawasan *resource* seperti CPU, memori, penyimpanan, dan layanan aktif, memungkinkan administrator mendeteksi anomali dan mengambil tindakan cepat untuk menjaga ketersediaan layanan [2].

2. Prometheus

Prometheus adalah alat *open-source* untuk *monitoring* dan *alerting* yang mengumpulkan data *time-series* dari target melalui HTTP. Dikembangkan pada 2012 dan menjadi bagian dari *Cloud Native Computing Foundation* sejak 2016, Prometheus menyimpan data dalam *time-series database* untuk analisis *real-time*, dengan kemampuan mendeteksi anomali melalui *alerting* [7].

3. Grafana

Grafana adalah *platform open-source* untuk visualisasi data metrik, terutama *time-series*. Alat ini mendukung berbagai sumber data seperti Prometheus dan menyediakan fitur *dashboard* interaktif serta *alerting* untuk notifikasi saat terjadi anomali, sehingga efektif untuk pemantauan sistem secara *real-time* [8].

4. Notifikasi Email

Notifikasi email digunakan dalam sistem *monitoring* untuk memberi tahu administrator tentang masalah seperti gangguan CPU, memori, atau layanan. Dengan notifikasi ini, administrator dapat merespons peringatan secara cepat untuk mencegah gangguan lebih lanjut [8].

5. Windows Exporter

Windows Exporter adalah alat yang mengumpulkan metrik dari server Windows, termasuk penggunaan CPU, memori, penyimpanan I/O, dan jaringan. Data ini dikirim ke Prometheus untuk analisis dan pemantauan *real-time* [9].

6. VMware vSphere Platform (ESXi)

VMware ESXi adalah *hypervisor tipe 1* yang memungkinkan beberapa mesin virtual berjalan pada satu perangkat fisik [10]. ESXi membantu mengelola *resource* seperti CPU, RAM, dan penyimpanan secara efisien, meningkatkan efisiensi operasional perusahaan [11].

7. Anomali Resource Server

Anomali *resource* server terjadi ketika penggunaan CPU, memori, penyimpanan, atau throughput menyimpang dari kondisi normal. Deteksi anomali penting untuk menjaga stabilitas sistem karena dapat menyebabkan penurunan performa atau *downtime* [12].

8. Heavy Load Test

Heavy Load Test adalah pengujian kinerja sistem di bawah beban tinggi untuk mengevaluasi kapasitas maksimum dan mengidentifikasi potensi masalah. Pengujian ini membantu mengoptimalkan sumber daya agar sistem tetap stabil saat lonjakan beban terjadi [13].

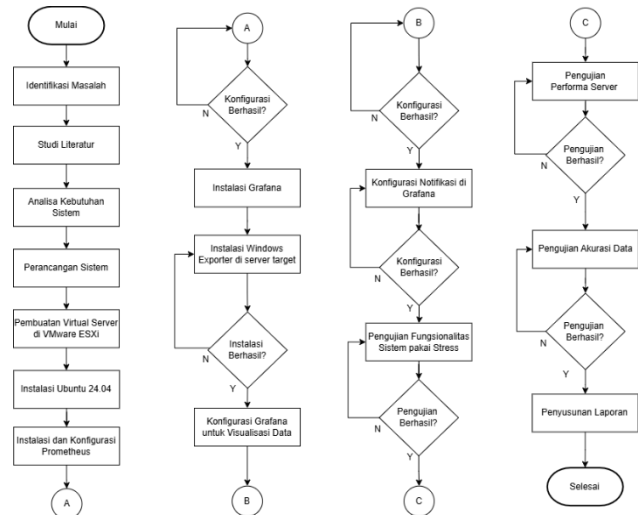
9. TamoSoft Throughput Test

TamoSoft Throughput Test adalah alat pengujian performa jaringan yang menghitung *throughput upstream* dan *downstream* menggunakan data TCP/UDP. Alat ini menampilkan hasil dalam format numerik dan grafik untuk mengevaluasi kinerja jaringan secara akurat [14].

III. METODOLOGI

A. Metode Penelitian

Metodologi penelitian ini melibatkan beberapa tahapan utama untuk merancang dan mengimplementasikan sistem *monitoring* server berbasis Prometheus dan Grafana. Perangkat lunak yang digunakan meliputi Ubuntu Server 24.04 sebagai sistem operasi virtual server, Prometheus 2.53.3 untuk pengumpulan dan penyimpanan metrik *time-series*, Grafana 11.3.1 untuk visualisasi data, Windows Exporter sebagai pengumpul metrik dari server Windows, email sebagai media notifikasi, VMware ESXi sebagai *hypervisor* virtualisasi, serta *Heavy Load Test* dan *TamoSoft Throughput Test* untuk pengujian beban. Alur metode penelitian ini dapat digambarkan pada diagram alir penelitian yang disajikan di Gambar 1.

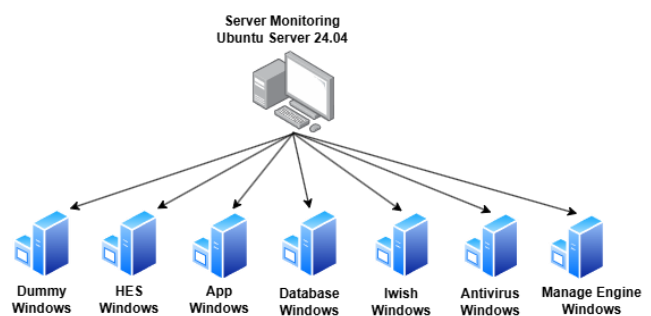


Gambar 1. Diagram Alir

Gambar 1 merupakan Tahapan penelitian dimulai dari identifikasi masalah, yaitu kesulitan administrator dalam memantau metrik server secara efisien dan kebutuhan akan sistem *monitoring* yang responsif. Selanjutnya dilakukan studi literatur untuk mengkaji referensi terkait alat dan metode *monitoring* server. Pada tahap perancangan, sistem *monitoring* dirancang dengan mengintegrasikan Prometheus, Grafana, Windows Exporter, dan sistem notifikasi email, termasuk analisis kebutuhan perangkat keras dan lunak. Tahap instalasi dan konfigurasi meliputi pembuatan virtual server di VMware ESXi, instalasi Ubuntu, Prometheus, Grafana, Windows Exporter, serta konfigurasi *dashboard* dan notifikasi pada Grafana. Terakhir, tahap pengujian dilakukan untuk mengevaluasi fungsionalitas, akurasi data *real-time*, dan performa sistem *monitoring* yang telah dibangun.

B. Desain Topologi Jaringan

Gambar 2 merupakan desain topologi jaringan dalam penelitian ini terdiri dari tujuh server Windows yang dimonitor oleh satu server Ubuntu 24.04 sebagai pusat pengumpulan data menggunakan Prometheus 2.53.3.



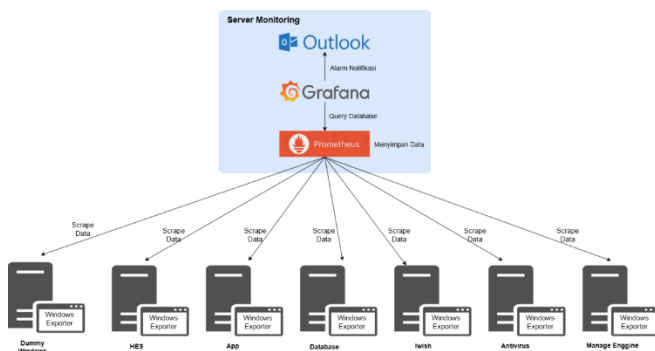
Gambar 2. Desain Topologi Jaringan

Setiap server Windows dilengkapi Windows Exporter untuk mengirim metrik kinerja CPU, memori, *disk* dan *throughput* ke Prometheus secara *real-time*, kemudian divisualisasikan melalui *dashboard* Grafana 11.3.1.

Sistem ini terintegrasi dengan mekanisme notifikasi email otomatis yang aktif saat terdeteksi anomali, memungkinkan respons cepat administrator. Seluruh infrastruktur diimplementasikan pada platform virtualisasi VMware ESXi untuk optimasi sumber daya.

C. Desain Sistem *Monitoring*

Gambar 3 merupakan desain sistem *monitoring* menggunakan Prometheus dan Grafana memantau server Windows dengan Windows Exporter yang mengumpulkan metrik CPU, memori, penyimpanan, dan jaringan, kemudian dikirim ke Prometheus melalui proses *scraping* dan disimpan dalam *database time-series*.



Gambar 3. Desain Sistem *Monitoring*

Grafana melakukan *query* data tersebut untuk divisualisasikan dalam *dashboard* interaktif, serta mengaktifkan *alert* dan mengirim notifikasi email melalui Outlook saat terdeteksi anomali. Sistem ini memudahkan administrator dalam pemantauan *real-time* dan respons cepat terhadap masalah pada *resource server*.

D. Skenario Pengujian

Pengujian sistem *monitoring* dilakukan melalui beberapa skenario untuk memastikan fungsionalitas, akurasi data, dan performa server. Pada pengujian fungsionalitas, dilakukan tiga skenario dengan memberikan beban pada server *dummy*, yaitu beban CPU, memori, dan *throughput*, masing-masing diuji sebanyak 10 kali untuk memastikan anomali terdeteksi di *dashboard* Grafana dan notifikasi email terkirim dengan baik. Selanjutnya, pengujian akurasi data *real-time* dilakukan dengan membandingkan metrik penggunaan CPU, memori, penyimpanan, dan *throughput* yang ditampilkan di Grafana dengan data referensi dari Task Manager atau *File Explorer* pada interval pengamatan 5 detik, guna memastikan tingkat akurasi dan presisi sistem. Terakhir, pengujian performa server dilakukan dengan memantau penggunaan *resource server monitoring* saat mengawasi berbagai jumlah host, mulai dari satu, dua, empat, hingga enam *host* selama 6 jam, untuk mengevaluasi efisiensi dan skalabilitas sistem dalam kondisi beban yang berbeda.

IV. HASIL DAN PEMBAHASAN

A. Tampilan Sistem

Tampilan *dashboard* Grafana PT HMMI disajikan pada Gambar 4.



Gambar 4. Tampilan *Dashboard* Grafana PT HMMI.

Dashboard Grafana di PT Hino Motors Manufacturing Indonesia memantau kinerja server Windows secara *real-time* dengan visualisasi metrik yang dikumpulkan oleh Windows Exporter melalui proses *scraping*. *Dashboard* ini terdiri dari empat panel utama seperti pada Gambar 4:

1. Panel *Throughput* untuk memantau lalu lintas data dan mendeteksi masalah jaringan atau serangan.
2. Panel *Hard Disk Usage* untuk mengawasi kapasitas penyimpanan guna mencegah kehabisan ruang.
3. Panel *Memory Usage* untuk mengidentifikasi penggunaan RAM yang berlebihan.
4. Panel *CPU Usage* untuk memantau beban prosesor. Sistem ini dilengkapi dengan fitur notifikasi email otomatis yang aktif saat terjadi anomali, membantu administrator merespons masalah dengan cepat.

B. Hasil Pengujian Fungsionalitas

Pengujian fungsionalitas dilakukan dalam tiga skenario untuk menguji keandalan sistem *monitoring* Grafana dalam memantau penggunaan CPU, memori, dan *throughput* secara *real-time* serta mengirim notifikasi email saat terjadi anomali, sementara pengujian penyimpanan tidak dilakukan karena Grafana hanya membaca kapasitas, bukan aktivitas *read/write disk*. Hasil pengujian fungsionalitas disajikan pada Tabel 1.

Tabel 1. Hasil Pengujian Fungsionalitas

Skenario	Status Berhasil	Notifikasi Berhasil	Persentase Berhasil
Skenario 1 CPU	10	10	100%
Skenario 2 Memori	10	10	100%
Skenario 3 Throughput	10	10	100%

Berdasarkan hasil rekap pada Tabel 1 pengujian fungsionalitas dari skenario pertama hingga ketiga diperoleh kesimpulan bahwa pada percobaan pemberian beban CPU, memori, dan *throughput* memiliki persentase keberhasilan 100% perubahan data dapat terbaca pada Grafana dan mampu mengirimkan notifikasi secara *real-time* saat terjadi anomali.

C. Hasil Pengujian Akurasi Data

Pengujian akurasi data dilakukan pada seluruh server dengan membandingkan metrik penggunaan CPU, memori, penyimpanan, dan *throughput* antara visualisasi Grafana dan *Task Manager*. Setiap pengujian berlangsung selama lima detik, sesuai dengan interval waktu terkecil yang dapat dipantau secara *real-time*, dan terdiri dari empat skenario yang masing-masing diuji sebanyak 30 kali. Hasil pengujian akurasi data disajikan pada Tabel 2.

Tabel 2. Hasil Pengujian Akurasi Data

Skenario	Jenis Beban	Jumlah Percobaan	Persentase Akurasi
Skenario 1	CPU	30	95,20%
Skenario 2	Memori	30	99,71%
Skenario 3	Disk	30	99,14%
Skenario 4	Throughput	30	91,42%

Berdasarkan hasil rekap pada Tabel 2 pengujian akurasi data dapat disimpulkan bahwa nilai akurasi data secara *real-time* pada *dashboard* Grafana pada penggunaan CPU mencapai 95,20%, penggunaan memori 99,71%, penggunaan *disk* 99,71%, dan penggunaan *throughput* senilai 91,42%.

D. Hasil Pengujian Performa Server

Pengujian performa dilakukan untuk mengukur besarnya sumber daya server yang digunakan dalam proses monitoring. Data yang diambil meliputi rata-rata penggunaan CPU, memori, *disk*, dan *throughput*. Pengujian ini dilakukan dalam empat skenario, yaitu monitoring terhadap satu, dua, empat, dan enam *client* server secara berturut-turut selama enam jam setiap skenario. Hasil pengujian performa server disajikan pada Tabel 3.

Tabel 3. Hasil Pengujian Performa Server

Skenario	Host	CPU	Memori	Disk	Throughput
Pertama	1	0,825%	22,4 MB	20,85 MB	72,95 Kbps
Kedua	2	0,97%	22,3 MB	20,72 MB	83,28 Kbps
Ketiga	4	1,11%	22,45 MB	20,55 MB	105,39 Kbps
Keempat	6	2,11%	22,48 MB	20,42 MB	203,56 Kbps

Berdasarkan Tabel 3, pengujian performa server menunjukkan rata-rata penggunaan CPU tertinggi 2,11%,

memori maksimum 22,48%, ruang *disk* terbesar 20,85%, dan *throughput* tertinggi 203,56 Kbps. Penggunaan CPU, *disk*, dan *throughput* meningkat seiring bertambahnya jumlah server yang di monitor.

V. SIMPULAN

Implementasi sistem *monitoring* server menggunakan Prometheus dan Grafana di PT HMMI berhasil memantau *resource* server dan mendeteksi anomali secara efektif, dengan tingkat keberhasilan notifikasi email mencapai 100%. Pengujian akurasi data *real-time* pada *dashboard* Grafana menunjukkan hasil akurasi tinggi, yaitu penggunaan CPU 95,20%, memori 99,71%, *disk* 99,71%, dan *throughput* 91,42%. Pengujian performa server mencatat penggunaan CPU tertinggi 2,11%, memori maksimum 22,48%, *disk* terbesar 20,85%, dan *throughput* tertinggi 203,56 kbps.

REFERENSI

- [1] W. Mahendra Ardiansyah, "Peran Teknologi dalam Transformasi Ekonomi dan Bisnis di Era Digital," 2023. [Online]. Available: <https://journal.sabajayapublisher.com/index.php/jmeh>
- [2] D. Rahman and H. Amnur, "Indri Rahmayuni 133 Monitoring Server dengan Prometheus dan Grafana serta Notifikasi Telegram Jurnal Ilmiah Teknologi Sistem Informasi," 2020. [Online]. Available: <http://jurnal-itsi.org>
- [3] J. Homepage, B. Rasyidi, and F. Pratama, "MALCOM: Indonesian Journal of Machine Learning and Computer Science Server Monitoring System at PT. XYZ Media Indonesia Based on Grafana and Prometheus Sistem Monitoring Server di PT. XYZ Media Indonesia Berbasis Grafana dan Prometheus," vol. 4, pp. 1456–1465, 2024, doi: 10.57152/malcom.v4i4.1546.
- [4] K. Anam, "IMPLEMENTASI SISTEM MONITORING RESOURCE SERVER BERBASIS PROMETHEUS DAN GRAFANA DENGAN NOTIFIKASI TELEGRAM".
- [5] A. Rahma, F. Indriyani, T. Alfian, and A. Sandi, "Perancangan Dan Implementasi Monitoring Perangkat Server Menggunakan Zabbix Pada PT. Rizki Tujuh Belas Kelola," Jurnal INSAN (Journal of Information Systems Management Innovation, vol. 3, no. 2, 2023, [Online]. Available: <http://jurnal.bsi.ac.id/index.php/jinsan>
- [6] A. Muhammad, "IMPLEMENTASI SISTEM MONITORING SERVER BERBASIS ANDROID MENGGUNAKAN OPENTELEMETRY DAN PROMETHEUS," 2024.
- [7] R. M. Febriana, "IMPLEMENTASI SISTEM MONITORING MENGGUNAKAN PROMETHEUS DAN GRAFANA."
- [8] R. ; Khairil and K. U. Suryana, "Security System Implementation And Monitoring Networks At Sma N 10 City Of Bengkulu," 2023.

-
- [9] herdiana3389, "Monitoring Windows With Windows Exporter Prometheus Grafana." Accessed: Feb. 09, 2025. [Online]. Available: <https://sys-ops.id/monitoring-windows-with-windows-exporter-prometheus-grafana/>
- [10] Zahiyah, "Apa Definisi VMware ESXi dan Pengaruhnya dalam Virtualisasi." Accessed: Mar. 19, 2025. [Online]. Available: <https://www.vinchin.com/id/vm-tips/apa-itu-vmware-esxi.html#:~:text=menambah%20banyak%20keunggulan.-,Perbedaan%20antara%20VMware%20ESXi%20dan%20VMware%20Workstation,lokal%20di%20PC%20melalui%20klien.>
- [11] R. A. Nuryadin, T. A. Ramadhani, J. Karaman, and M. Reza, "METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi ANALISIS PERBANDINGAN PERFORMA VIRTUALISASI SERVER MENGGUNAKAN VMWARE ESXI, ORACLE VIRTUAL BOX, VMWARE WORKSTATION 16 DAN PROXMOX," vol. 7, no. 2, 2023, doi: 10.46880/jmika.Vol7No2.pp175-180.
- [12] A. Asmarani, "KLASIFIKASI ANOMALI TRAFIK PADA IDS MENGGUNAKAN ALGORITMA NAÏVE BAIYES DAN RANDOM FOREST PROPOSAL TUGAS AKHIR," 2022.
- [13] "Alat Uji Stress Gratis Heavy Load." Accessed: Mar. 19, 2025. [Online]. Available: https://www-jam--software-com.translate.goog/heavyload?_x_tr_sl=en&_x_tr_tl=id&_x_tr_hl=id&_x_tr_pto=sge#:~:text=Heavyload%20is%20a%20System%20Utilities,draws%20patterns%20in%20its%20window.
- [14] "Throughput Test." Accessed: Mar. 19, 2025. [Online]. Available: <https://www.tamos.com/htmlhelp/tt/index.html>
-