

IMPLEMENTASI SISTEM MONITORING MENGGUNAKAN PROMETHEUS DAN GRAFANA



Kelompok 10:

Yasir Maarif (3122600013)

Abdul Khalim Efendi (3122600019)

Saiq Syahru Qadri (3122600029)

POLITEKNIK ELEKTRONIKA NEGERI SURABAYA

A. ABSTRAKSI

Pemantauan jaringan komputer sangat penting dilakukan agar memudahkan seorang system administrator jaringan dalam mengontrol system jaringan yang ada. Kebutuhan jaringan yang semakin besar membuat resiko gangguan hingga kerusakan dalam suatu jaringan juga semakin besar. Karena itu seorang system administrator diharuskan selalu memantau seluruh system jaringan. Ada beberapa langkah yang dilakukan saat melakukan perancangan system monitoring yang diinginkan. Prometheus dan Grafana dapat membantu membuat system monitoring jaringan yang mudah dimengerti dan dapat membantu seorang system administrator. Prometheus akan mengambil data dari protocol SNMP dan memproses data tersebut menggunakan exporter. Untuk melengkapi system monitoring ini diperlukan visualisasi. Grafana dapat membuat visualisasi dari data yang telah diproses. Hasil penelitian in menunjukkan device yang terhubung didalam jaringan dapat ditampilkan grafana.

Kata Kunci – Prometheus, Node Exporter, Grafana

B. PENDAHULUAN

Seiring dengan perkembangan teknologi, banyak perusahaan mulai memanfaatkan Virtual Machine (VM) untuk meningkatkan fleksibilitas dan efisiensi infrastruktur IT mereka. Namun, penggunaan VM dalam skala besar membutuhkan sistem monitoring yang handal untuk memastikan performa dan ketersediaan tetap optimal.

Prometheus dan Grafana adalah dua alat yang sering digunakan untuk keperluan ini. Prometheus adalah sistem monitoring open-source yang mengumpulkan dan menyimpan metrik dari berbagai sumber dalam format time-series. Sementara itu, Grafana adalah platform visualisasi yang memungkinkan pengguna untuk membuat dashboard interaktif untuk memantau data yang dikumpulkan oleh Prometheus.

Implementasi monitoring dengan Prometheus dan Grafana pada VM memberikan beberapa keuntungan utama:

1. Visibilitas: Dengan menggunakan Prometheus dan Grafana, perusahaan dapat memantau performa dan kesehatan VM secara real-time.

2. Deteksi Masalah Cepat: Data yang dikumpulkan memungkinkan deteksi masalah performa lebih cepat, sehingga dapat segera diatasi sebelum menjadi kritis.
3. Visualisasi Data: Dashboard Grafana yang interaktif membantu dalam memahami tren dan anomali dengan lebih mudah.
4. Open-Source dan Skalabilitas: Kedua alat ini adalah open-source dan dapat diskalakan sesuai kebutuhan, memberikan fleksibilitas dan mengurangi biaya.

Dengan latar belakang ini, implementasi monitoring menggunakan Prometheus dan Grafana menjadi langkah penting dalam menjaga kinerja optimal dan stabilitas lingkungan virtual perusahaan.

C. RUANG LINGKUP

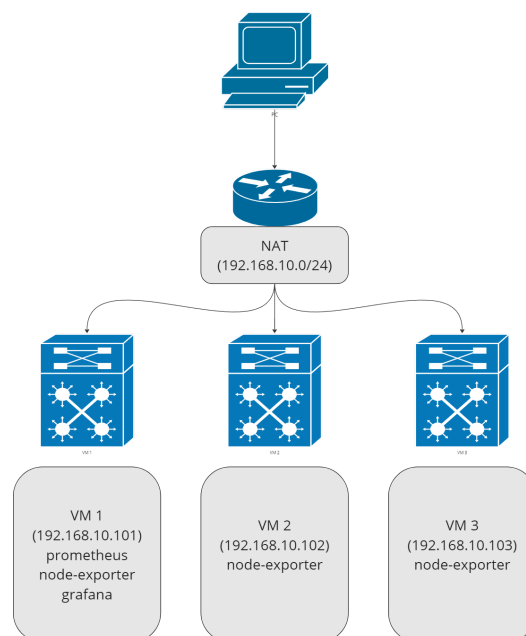
1. Monitoring Infrastruktur VM
 - Proyek ini berfokus pada monitoring tiga Virtual Machines (VM) yang berjalan di VirtualBox.
 - Masing-masing VM dikonfigurasi dengan IP statis dalam NAT Network.
2. Komponen Monitoring
 - Prometheus: Digunakan untuk mengumpulkan dan menyimpan data metrik dari VM.
 - Node Exporter: Diinstal di setiap VM untuk mengeksport metrik sistem seperti CPU, memori, disk, dan jaringan.
 - Grafana: Digunakan untuk visualisasi data metrik yang dikumpulkan oleh Prometheus.
3. Monitoring Metrics
 - CPU Usage: Memantau persentase penggunaan CPU.
 - Memory Usage: Memantau penggunaan RAM dan SWAP.
 - Disk Usage: Memantau penggunaan filesystem root.
 - System Load: Memantau load sistem keseluruhan.
 - Network Traffic: Memantau lalu lintas jaringan.
4. Visualisasi dan Dashboard

- Membuat dashboard di Grafana dengan berbagai panel untuk menampilkan metrik yang dikumpulkan.
- Menyediakan visualisasi metrik seperti penggunaan CPU, memori, disk, dan beban sistem secara real-time.

5. Tujuan Proyek

- Memastikan bahwa semua VM terpantau dengan baik.
- Mengidentifikasi potensi masalah kinerja atau beban pada VM.
- Menyediakan data yang bisa digunakan untuk analisis lebih lanjut mengenai performa VM.

D. DESAIN SISTEM



1. Grafana

- Fungsi: Grafana adalah platform visualisasi yang digunakan untuk menampilkan metrik yang dikumpulkan oleh Prometheus. Grafana memungkinkan Anda untuk membuat dashboard interaktif yang membantu dalam memantau dan menganalisis data.
- Interaksi: Grafana melakukan query ke Prometheus untuk mendapatkan data metrik yang kemudian ditampilkan dalam bentuk grafis di dashboard.

2. Prometheus Server

- Fungsi: Prometheus bertindak sebagai server pengumpulan metrik. Ini adalah pusat pengumpulan data yang mendapatkan data metrik dari berbagai sumber (dalam hal ini dari Node Exporter di setiap VM).
- Interaksi: Prometheus melakukan scraping terhadap Node Exporter yang berjalan di masing-masing VM untuk mengumpulkan data metrik. Data ini kemudian disimpan di database internal Prometheus.
- Data Collection: Prometheus dikonfigurasi untuk secara periodik mengakses endpoint dari Node Exporter yang berjalan di setiap VM untuk mengumpulkan metrik.

3. VM1, VM2, VM3

- Fungsi: Setiap VM menjalankan Node Exporter yang mengumpulkan dan mengekspor data metrik sistem seperti penggunaan CPU, memori, disk, dan jaringan.
- Interaksi: Node Exporter pada masing-masing VM menyediakan endpoint yang dapat diakses oleh Prometheus untuk scraping data metrik. Endpoint ini berada di port 9100

E. TIM dan TUGAS

Daftar anggota tim:

1. Yasir Maarif (3122600013)
2. Abdul Khalim Efendi (3122600019)
3. Saiq Syahru Qadri (3122600029)

Eksekutor	Tugas
Fendy, Saiq, Yasir	Instalasi dan set up virtual machine
Yasir, Saiq	Membuat project charter
Saiq, Fendy	Set up NAT Network dan IP
Yasir	Konfigurasi Prometheus, node exporter, grafana
Yasir, Saiq, Fendy	Menganalisa hasil monitoring di grafana
Yasir, Fendy	Membuat laporan akhir

F. TAHAPAN PELAKSANAAN

Nomor	Tanggal	Keterangan
1.	Senin, 27 Mei 2024	Membuat project charter
2.	Selasa, 28 Mei 2024	Setup 3 virtual machine dan networknya
3.	Jumat, 31 Mei 2024	Install node exporter di semua virtual machine
4.	Sabtu, 1 Juni 2024	Mengamati hasil monitoring dengan Prometheus dan Grafana
5.	Minggu, 2 Juni 2024	Membuat laporan akhir

G. IMPLEMENTASI

1. Persiapan Lingkungan

- Membuat Virtual Machine (VM) di VirtualBox
- Buat tiga VM dengan konfigurasi berikut:
 - a. VM1: 192.168.10.101 (Prometheus, Node Exporter, Grafana)
 - b. VM2: 192.168.10.102 (Node Exporter)
 - c. VM3: 192.168.10.103 (Node Exporter)
- Set Network Adapter menjadi NAT Network dan assign IP static pada masing-masing VM.
- Gateway: 192.168.10.1
- Konfigurasi Jaringan
- Pastikan setiap VM bisa saling ping untuk memastikan jaringan NAT Network berjalan dengan baik.

2. Instalasi dan Konfigurasi Prometheus di VM1

- Instalasi Prometheus
- Download Prometheus dari situs resminya:
- Ekstrak dan pindahkan ke direktori yang diinginkan:
- Buat user khusus untuk Prometheus:
- Konfigurasi Prometheus
- Edit file konfigurasi Prometheus (prometheus.yml) untuk menambahkan target monitoring.
- Menjalankan Prometheus
- Buat service file untuk Prometheus

- Start dan enable service Prometheus.
3. Instalasi dan Konfigurasi Node Exporter di Semua VM
- Instalasi Node Exporter
 - Download Node Exporter dari situs resminya
 - Ekstrak dan pindahkan ke direktori yang diinginkan
 - Menjalankan Node Exporter
 - Buat service file untuk Node Exporter.
 - Start dan enable service Node Exporter
4. Instalasi dan Konfigurasi Grafana di VM1
- Instalasi Grafana
 - Tambahkan repository Grafana dan install
 - Menjalankan Grafana
 - Start dan enable service Grafana
 - Konfigurasi Grafana
 - Akses Grafana melalui browser pada <http://192.168.10.101:3000>.
 - Login dengan default user dan password (admin/admin).
 - Tambahkan data source Prometheus.
 - a. Name: Prometheus
 - b. Type: Prometheus
 - c. URL: <http://192.168.10.101:9090>
 - d. Save & Test
 - Buat dashboard dan tambahkan panel untuk memonitoring masing-masing VM menggunakan metrics dari Node Exporter.
5. Validasi dan Pengujian
- Cek Status Prometheus dan Node Exporter
 - Akses Prometheus melalui browser pada <http://192.168.10.101:9090> dan pastikan target monitoring (VM2 dan VM3) terdeteksi.
 - Akses Node Exporter dari masing-masing VM pada port 9100 dan pastikan metrics tersedia.
 - Monitoring dengan Grafana
 - Buat dashboard dengan panel untuk memonitor CPU, Memory, Disk, dan Network usage dari ketiga VM.

- Simpan dan lakukan pengujian untuk memastikan data metrics terbaca dengan benar di Grafana.
6. Dokumentasi
- Buat dokumentasi yang mencakup semua langkah instalasi dan konfigurasi di atas, beserta screenshot dari hasil monitoring di Grafana.

H. SISTEM TESTING

1. Pengujian Prometheus

Pengujian Prometheus ini dilakukan setelah instalasi dan konfigurasi selesai dilakukan. Pengujian ini dilakukan agar memastikan bahwa prometheus telah aktif. Tambahkan konfigurasi target VM dalam file `prometheus.yml` pada bagian `scrape_configs`. Jika berhasil melakukan proses instalasi dan konfigurasi Prometheus, saat memanggil localhost:9090 maka akan terakses jendela Prometheus.

2. Verifikasi Pengumpulan Metrik dari VM

Setelah pengujian awal yang harus dilakukan adalah me-restart dan membuka kembali halaman Prometheus di <http://localhost:9090>. Dengan menggunakan tab "Targets" untuk memastikan bahwa VM yang dikonfigurasi muncul sebagai target yang aktif dan ter-scrape. Jika target VM muncul dengan status "UP", berarti Prometheus berhasil mengumpulkan metrik dari VM.

3. Setup Grafana untuk Visualisasi Data

Akses Grafana melalui URL <http://localhost:3000> setelah instalasi dan konfigurasi. Tambahkan Prometheus sebagai data source di Grafana: Masuk ke Grafana, lalu pergi ke Configuration > Data Sources. Klik "Add data source" dan pilih "Prometheus". Isi URL Prometheus (misalnya <http://localhost:9090>) dan klik "Save & Test" untuk memastikan koneksi berhasil. Buat dashboard baru untuk memvisualisasikan metrik VM: Masuk ke Create > Dashboard. Tambahkan panel baru dan pilih data source Prometheus. Masukkan query yang sesuai untuk metrik VM yang ingin dipantau, seperti `up{job="virtual_machines"}` untuk melihat status VM.

4. Verifikasi Visualisasi Data di Grafana

Pastikan metrik dari VM muncul di dashboard Grafana. Periksa apakah data yang ditampilkan sesuai dengan metrik yang diharapkan, seperti penggunaan CPU, memory, disk I/O, dan lain-lain.

5. Pengujian End-to-End

Melakukan beberapa aktivitas pada VM (misalnya, beban kerja CPU tinggi atau aktivitas jaringan) untuk melihat apakah perubahan tersebut tercermin dalam metrik yang dikumpulkan dan divisualisasikan. Kemudian memantau dashboard Grafana untuk memastikan bahwa metrik yang ditampilkan real-time dan sesuai dengan aktivitas yang dilakukan pada VM.

