DevOps

Day 6

Date: 22.03.2025

Topics Covered: Prometheus, Grafana and Java Application Minikube Deployement

Prometheus and Grafana

Prometheus and Grafana are monitoring tools

Prometheus:

sudo useradd \

  --system \

  --no-create-home \

  --shell /bin/false Prometheus

wget https://github.com/prometheus/prometheus/releases/download/v2.47.1/prometheus-2.47.1.linux-amd64.tar.gz

tar -xvf prometheus-2.47.1.linux-amd64.tar.gz

sudo mkdir -p /data /etc/prometheus

cd prometheus-2.47.1.linux-amd64/

sudo mv prometheus promtool /usr/local/bin/

sudo mv consoles/ console_libraries/ /etc/prometheus/

sudo mv prometheus.yml /etc/prometheus/prometheus.yml

sudo chown -R prometheus:prometheus /etc/prometheus/ /data/

cd

rm -rf prometheus-2.47.1.linux-amd64.tar.gz

prometheus --version

sudo vim /etc/systemd/system/prometheus.service

[Unit]

Description=Prometheus

Wants=network-online.target

After=network-online.target

StartLimitIntervalSec=500

StartLimitBurst=5

```
[Service]
User=prometheus
Group=prometheus
Type=simple
Restart=on-failure
RestartSec=5s
ExecStart=/usr/local/bin/prometheus \
  --config.file=/etc/prometheus/prometheus.yml \
  --storage.tsdb.path=/data \
  --web.console.templates=/etc/prometheus/consoles \
  --web.console.libraries=/etc/prometheus/console_libraries \
  --web.listen-address=0.0.0.0:9090 \
  --web.enable-lifecycle
[Install]
WantedBy=multi-user.target sudo systemctl enable prometheus
sudo systemctl start prometheus
sudo systemctl status prometheus
journalctl -u prometheus -f --no-pager
sudo useradd \
    --system \
    --no-create-home \
    --shell /bin/false node_exporter
wget https://github.com/prometheus/node_exporter/releases/download/v1.6.1/node_exporter-1.6.1.linux-amd64.tar.gz
tar -xvf node_exporter-1.6.1.linux-amd64.tar.gz
sudo mv \
  node_exporter-1.6.1.linux-amd64/node_exporter \
  /usr/local/bin/
rm -rf node_exporter*
node_exporter –version
sudo vim /etc/systemd/system/node_exporter.service
```

```
[Unit]
Description=Node Exporter
Wants=network-online.target
After=network-online.target
StartLimitIntervalSec=500
StartLimitBurst=5
[Service]
User=node_exporter
Group=node_exporter
Type=simple
Restart=on-failure
RestartSec=5s
ExecStart=/usr/local/bin/node_exporter \
  --collector.logind
[Install]
WantedBy=multi-user.target
sudo systemctl enable node_exporter
sudo systemctl start node_exporter
sudo systemctl status node_exporter
journalctl -u node_exporter -f --no-pager
sudo vim /etc/prometheus/prometheus.yml
- job_name: 'jenkins'
  metrics_path: '/prometheus'
  static_configs:
   - targets: ['<jenkins-ip>:8080']
promtool check config /etc/prometheus/prometheus.yml
curl -X POST http://localhost:9090/-/reload
sudo apt-get install -y apt-transport-https software-properties-common
wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add -
echo "deb https://packages.grafana.com/oss/deb stable main" | sudo tee -a
/etc/apt/sources.list.d/grafana.list
```

sudo apt-get update

sudo apt-get -y install grafana

sudo systemctl enable grafana-server

sudo systemctl start grafana-server

sudo systemctl status grafana-server

Grafana:

Open source

Prometheus is an open-source system monitoring and alerting toolkit originally built at SoundCloud. It is now a standalone open source project . Prometheus joined the Cloud Native Computing Foundation in 2016 as the second hosted project, after Kubernetes.

Features:

multi dimensional

Features

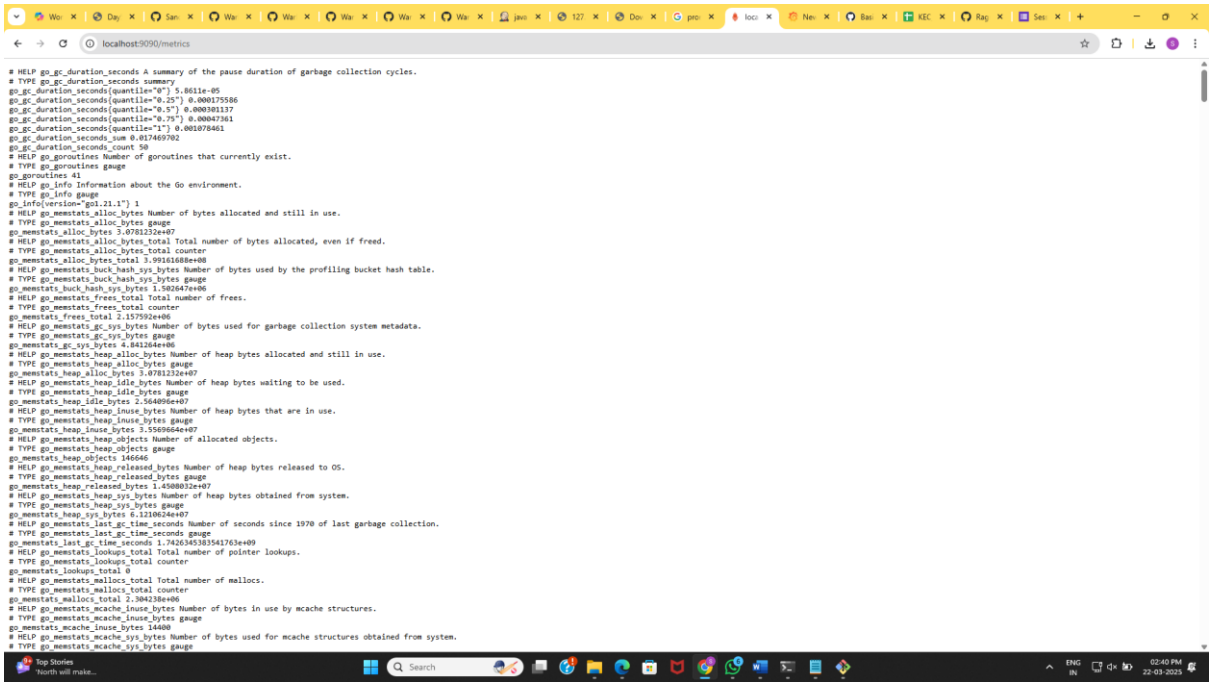
1. a multi-dimensional data model with time series data identified by metric name and key/value pairs
2. PromQL, a flexible query language to leverage this dimensionality
3. no reliance on distributed storage; single server nodes are autonomous
4. time series collection happens via a pull model over HTTP
5. pushing time series is supported via an intermediary gateway
6. targets are discovered via service discovery or static configuration
7. multiple modes of graphing and dashboarding support

In Grafana new Dashboard

9964 - prometheus

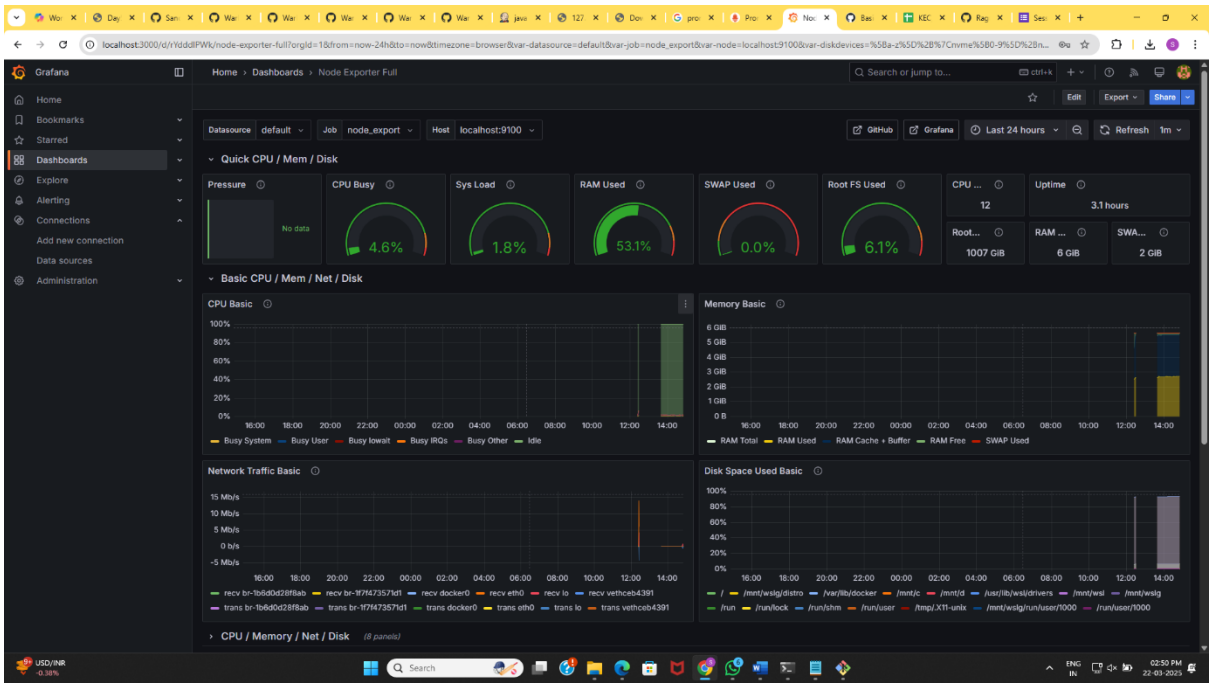
1960

Prometheus Metrics



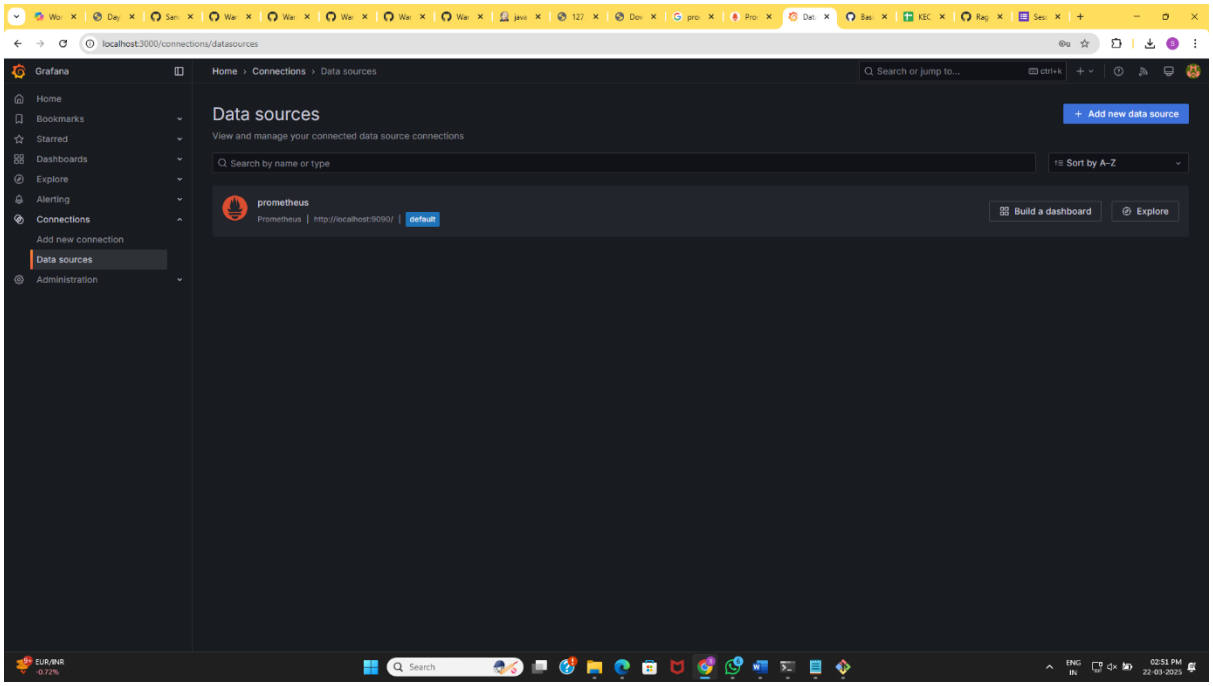
metrics

Grafana Prometheus Dashboard



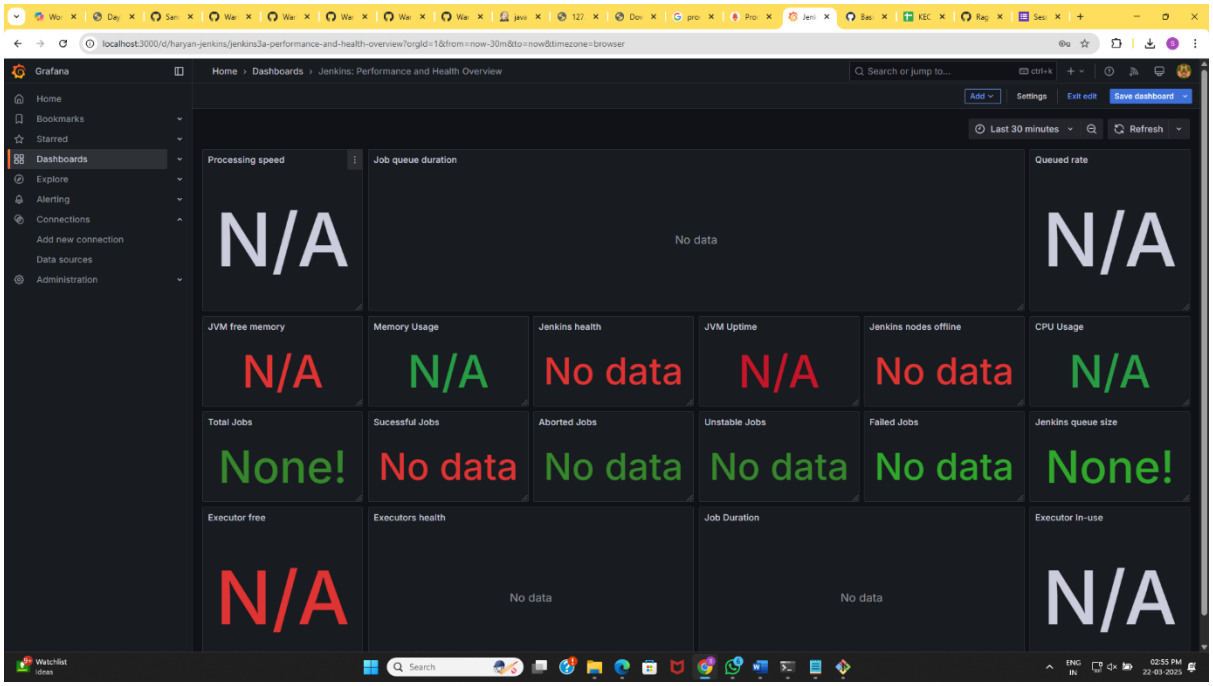
Grafana Dashboard

Creating Connections for Prometheous



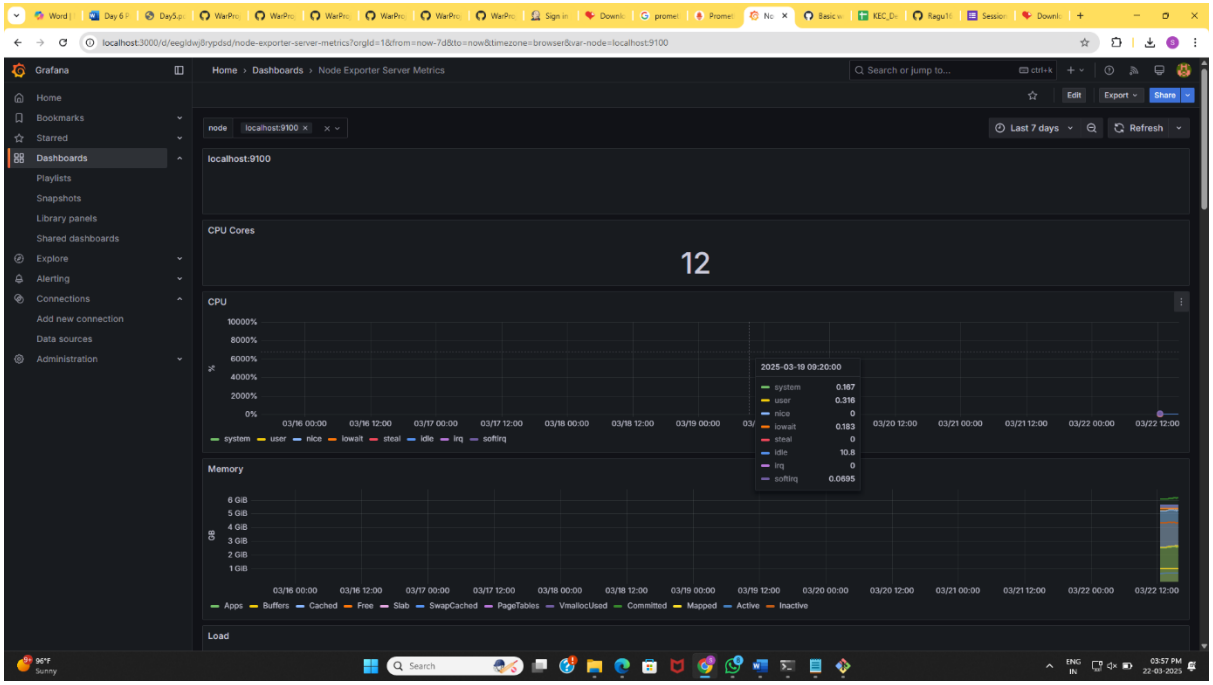
Connections

Jenkins Dashboard



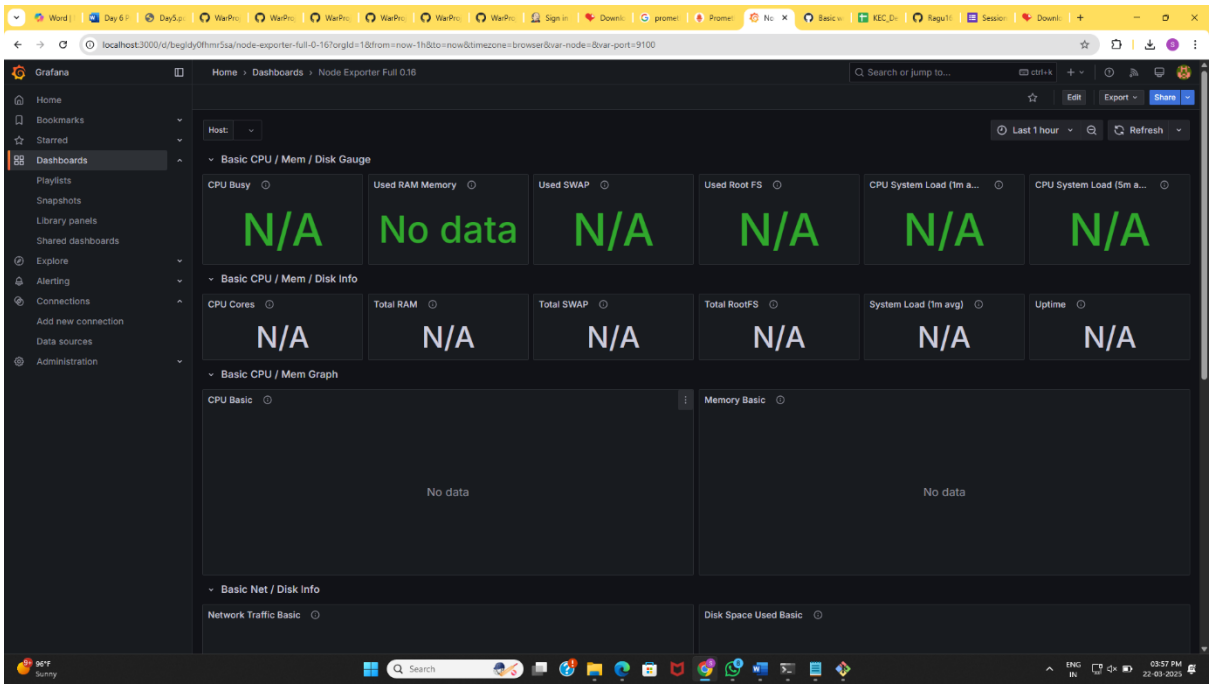
Grafana Dashboard

Code 504



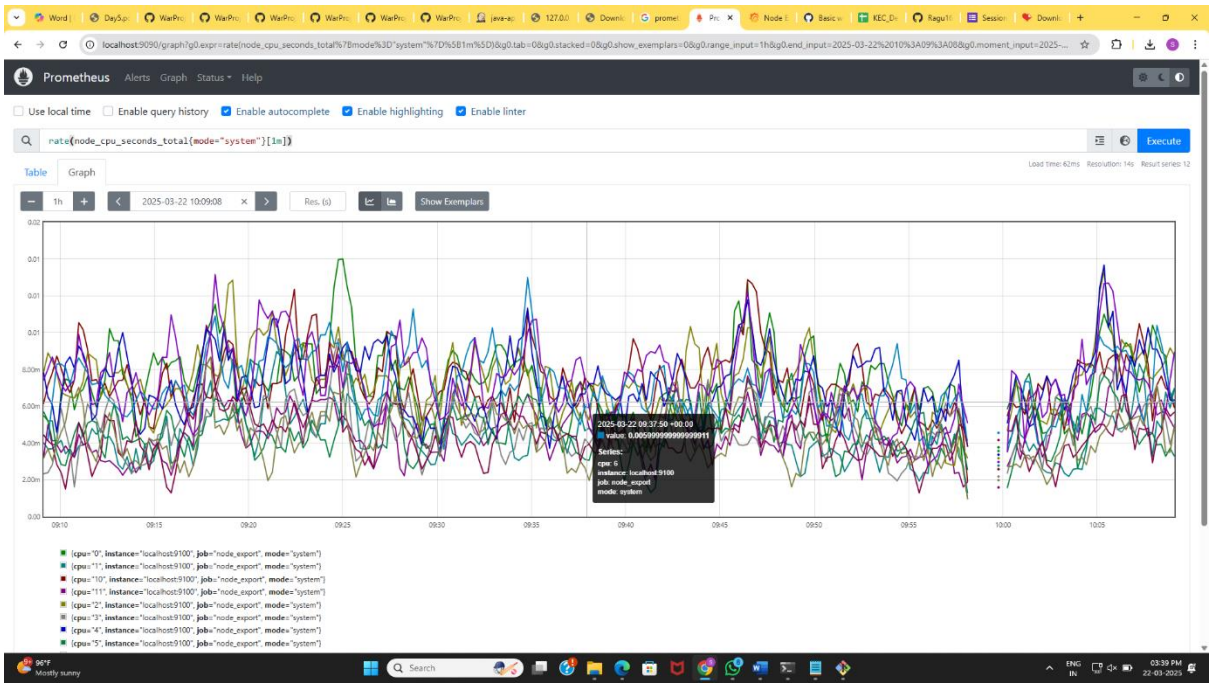
Grafana Dashboard

Code 5174



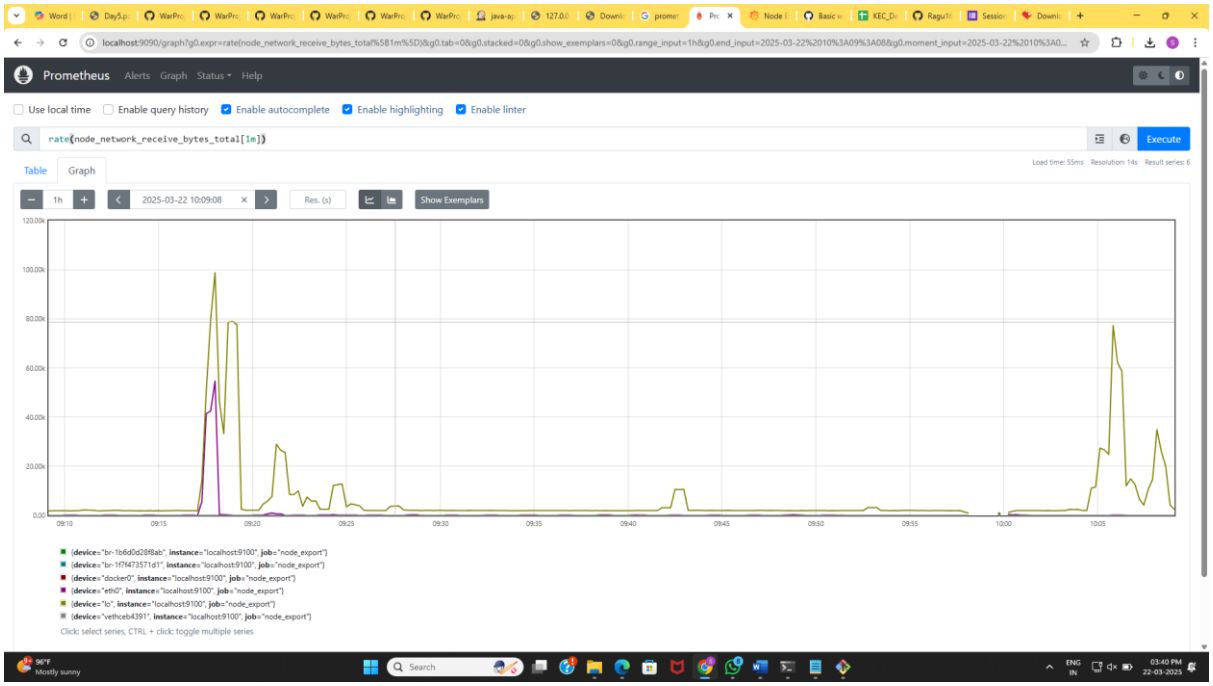
Grafana Dashboard

rate(node_cpu_seconds_total{mode="system"}[1m])



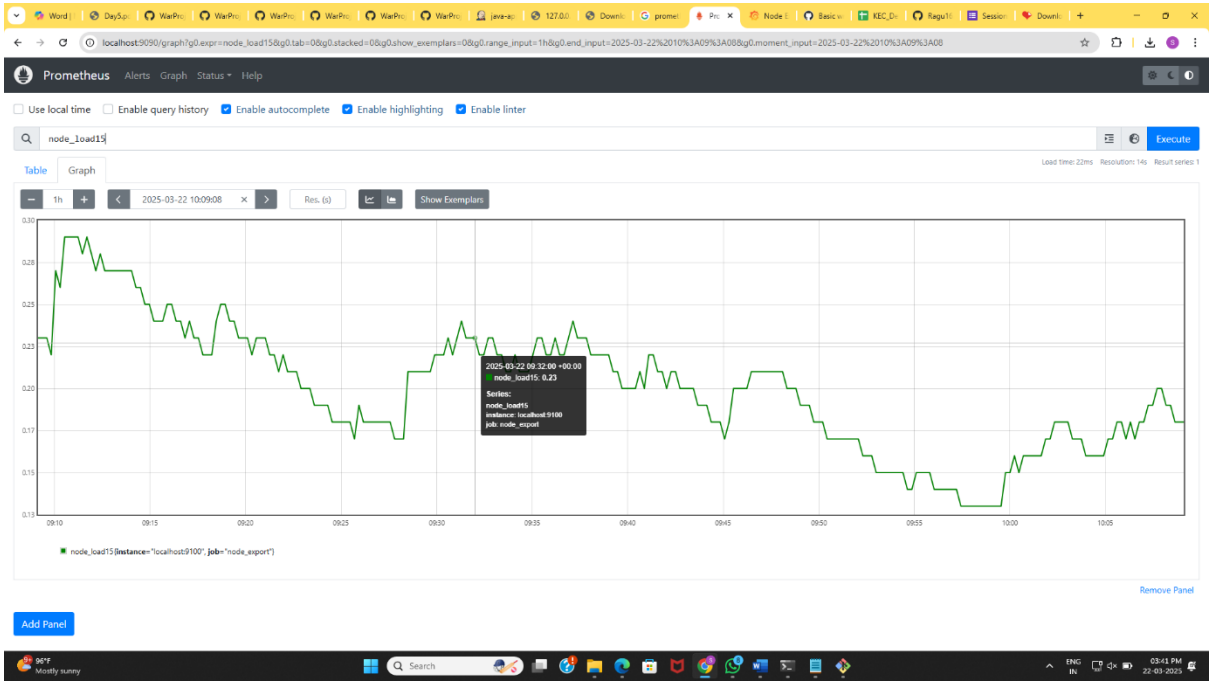
Prometheus

rate(node_network_receive_bytes_total[1m])



Prometheus

node_load15



Prometheus