

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,**

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

### **Prometheus Installation:**

Username Creation:

```
sudo useradd \  
--system \  
--no-create-home \  
--shell /bin/false Prometheus
```

### **Commands:**

```
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

prometh47.1/Prometheus-2.47.1.linux --version

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

```

```

naren@DESKTOP-3ELNUUV:~$ minikube start
minikube v1.35.0 on Ubuntu 24.04 (amd64)
Using the docker driver based on existing profile
Starting "minikube" primary control-plane node in "minikube" cluster
Pulling base image v0.0.46 ...
Updating the running docker "minikube" container ...
Failing to connect to https://registry.k8s.io/ from both inside the minikube container and host machine
To pull new external images, you may need to configure a proxy: https://minikube.sigs.k8s.io/docs/reference/networking/proxy/
Preparing Kubernetes v1.32.0 on Docker 27.4.1 ...
Verifying Kubernetes components...
  * Using image gcr.io/k8s-minikube/storage-provisioner:v5
  * Enabled addons: default-storageclass, storage-provisioner
Done! kubectcl is now configured to use "minikube" cluster and "default" namespace by default
naren@DESKTOP-3ELNUUV:~$ sudo useradd \
> --system \
> --no-create-home \
> shell /bin/false prometheus
[sudo] password for naren:
Usage: useradd [options] LOGIN
useradd -D
useradd -D [options]

Options:
--badname          do not check for bad names
-b, --base-dir BASE_DIR  base directory for the home directory of the
                        new account
--btrfs-subvolume-home  use BTRFS subvolume for home directory
-c, --comment COMMENT  GECOS field of the new account
-d, --home-dir HOME_DIR  home directory of the new account
-D, --defaults          print or change default useradd configuration
-e, --expiredate EXPIRE_DATE  expiration date of the new account
-f, --inactive INACTIVE  password inactivity period of the new account
-F, --add-subids-for-system  add entries to sub[uid] even when adding a system user
-g, --gid GROUP          name or ID of the primary group of the new
                        account
-G, --groups GROUPS      list of supplementary groups of the new
                        account
-h, --help              display this help message and exit
-k, --skel SKEL_DIR     use this alternative skeleton directory
-K, --key KEY=VALUE      override /etc/login.defs defaults
-l, --no-log-init        do not add the user to the lastlog and

4.tar.gz
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/6838921/2f9b7b37-63a0-428b-adb5-0294482fd743?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250322%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250322T062828Z&X-Amz-Expires=300&X-Amz-Signature=69fdd89440900928bf246f442aa53b1a8b9132cb2257c36be1bc990f5908a7f76&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dprometheus-2.47.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream [following]
--2025-03-22 06:28:32-- https://objects.githubusercontent.com/github-production-release-asset-2e65be/6838921/2f9b7b37-63a0-428b-adb5-0294482fd743?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20250322%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20250322T062828Z&X-Amz-Expires=300&X-Amz-Signature=69fdd89440900928bf246f442aa53b1a8b9132cb2257c36be1bc990f5908a7f76&X-Amz-SignedHeaders=host&response-content-disposition=attachment%3B%20filename%3Dprometheus-2.47.1.linux-amd64.tar.gz&response-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.111.133, 185.199.110.133, 185.199.109.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.111.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 95713066 (91M) [application/octet-stream]
Saving to: 'prometheus-2.47.1.linux-amd64.tar.gz'

prometheus-2.47.1.linux-amd64 100%[=====] 91.28M 602KB/s in 3m 28s

2025-03-22 06:32:10 (450 KB/s) - 'prometheus-2.47.1.linux-amd64.tar.gz' saved [95713066/95713066]

prometheus-2.47.1.linux-amd64/
prometheus-2.47.1.linux-amd64/LICENSE
prometheus-2.47.1.linux-amd64/NOTICE
prometheus-2.47.1.linux-amd64/prometheus.yml
prometheus-2.47.1.linux-amd64/consoles/

```

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 \

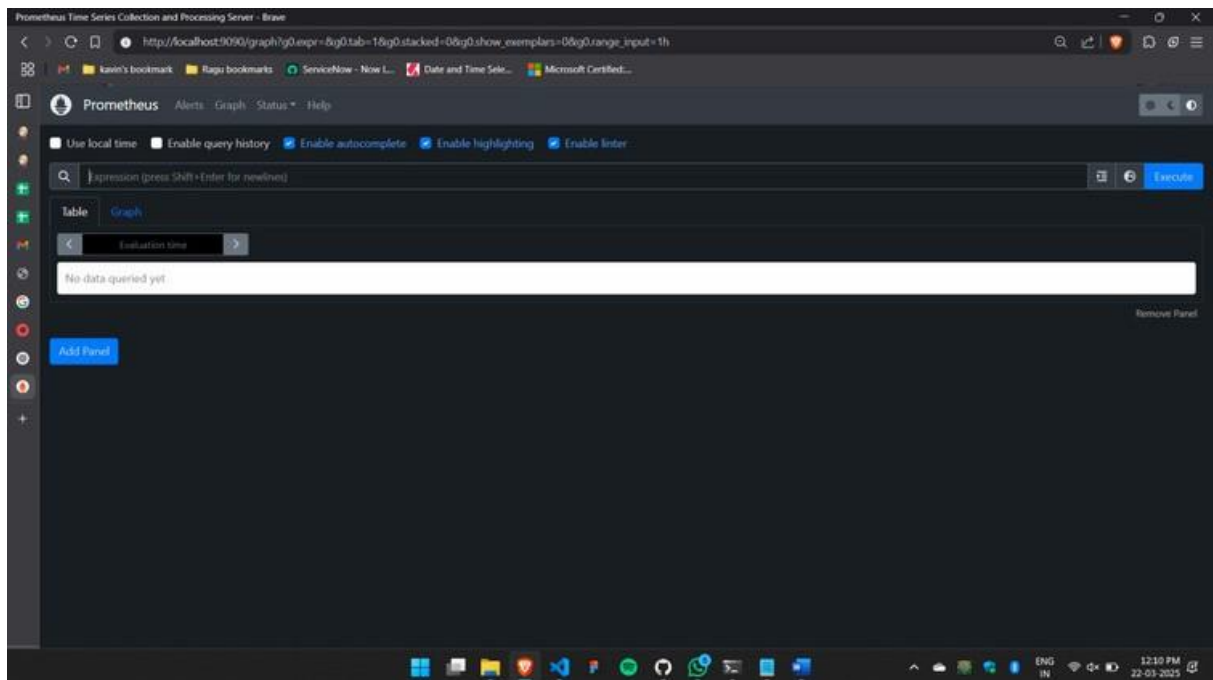
--web.enable-lifecycle

## [Install]

WantedBy=multi-user.target

```
ESC0D
prometheus.service - Prometheus
  Loaded: loaded (/etc/systemd/system/prometheus.service; enabled; preset: enabled)
  Active: active (running) since Sat 2025-03-22 06:40:02 UTC; 22ms ago
  Main PID: 40517 (prometheus)
  Tasks: 5 (limit: 8240)
  Memory: 2.8M ()
  CGroup: /system.slice/prometheus.service
          └─40517 /usr/local/bin/prometheus --config.file=/etc/prometheus/prometheus.yml --storage.tsdb.path=/data

Mar 22 06:40:02 DESKTOP-ELKSV5N systemd[1]: Started prometheus.service - Prometheus.
```



Node Exporter:

Commands for installation:

```
sudo mv \
    node_exporter-1.6.1.linux-amd64/node_exporter \
    /usr/local/bin/
rm -rf node_exporter*
```

Node exporter file:

[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

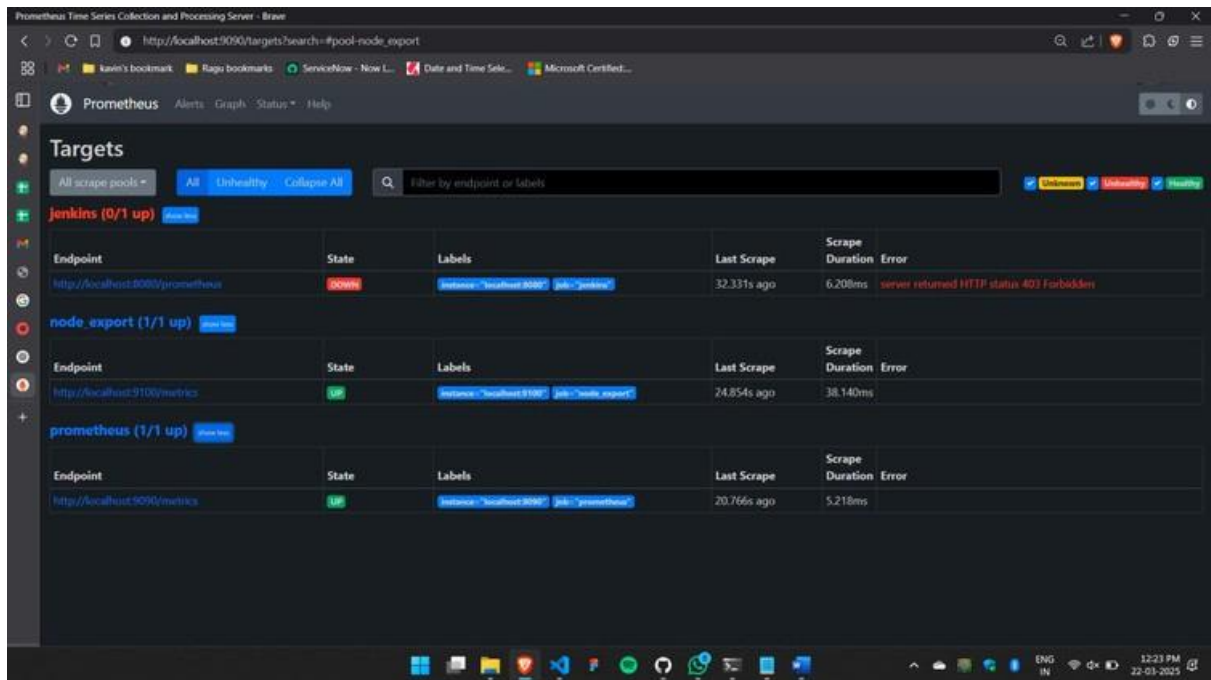
Service check : sudo systemctl enable node\_exporter

sudo systemctl start node\_exporter

sudo systemctl status node\_exporter

journalctl -u node\_exporter -f --no-pager

Reload Prometheus: curl -X POST <http://localhost:9090/-/reload>



## Grafana:

```
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
```

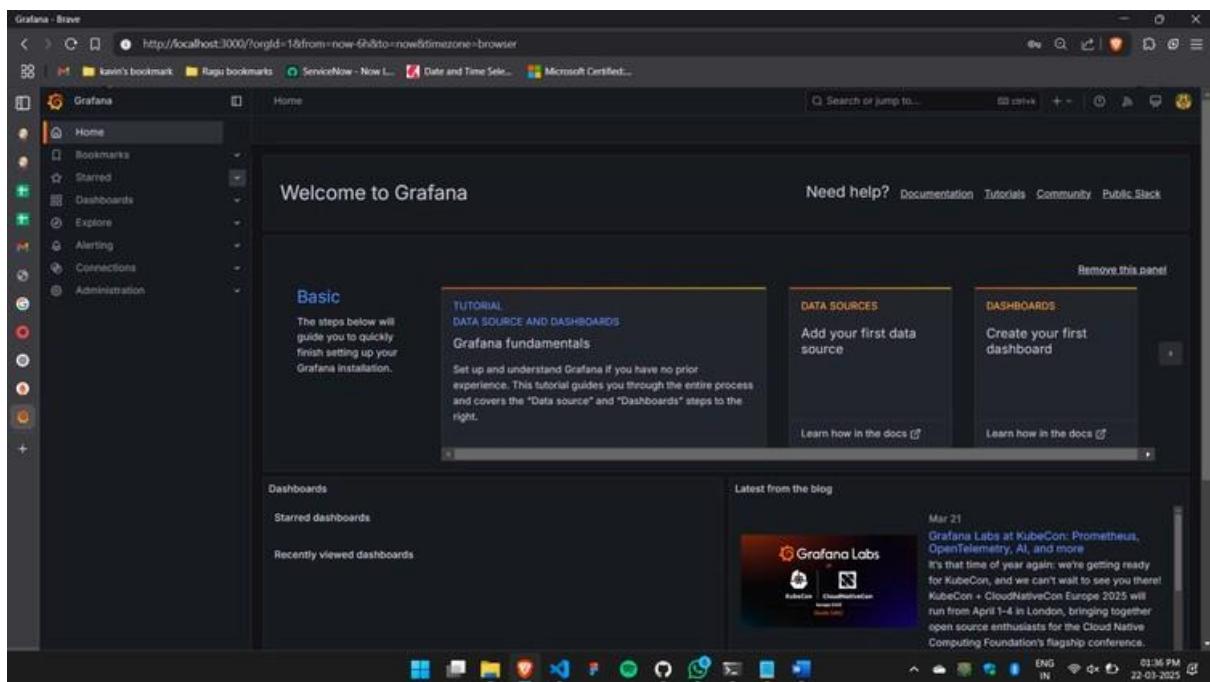
```
sudo apt-get -y install grafana
```

```
sudo systemctl enable grafana-server
```

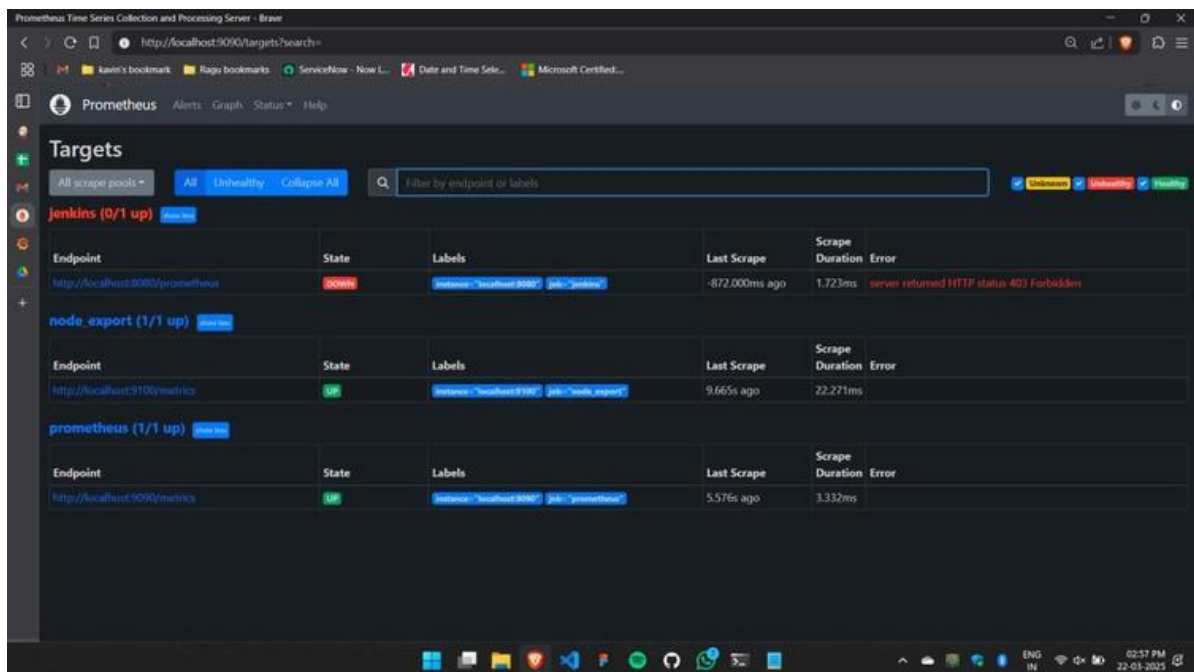
```
sudo systemctl start grafana-server
```

```
sudo systemctl status grafana-server
```

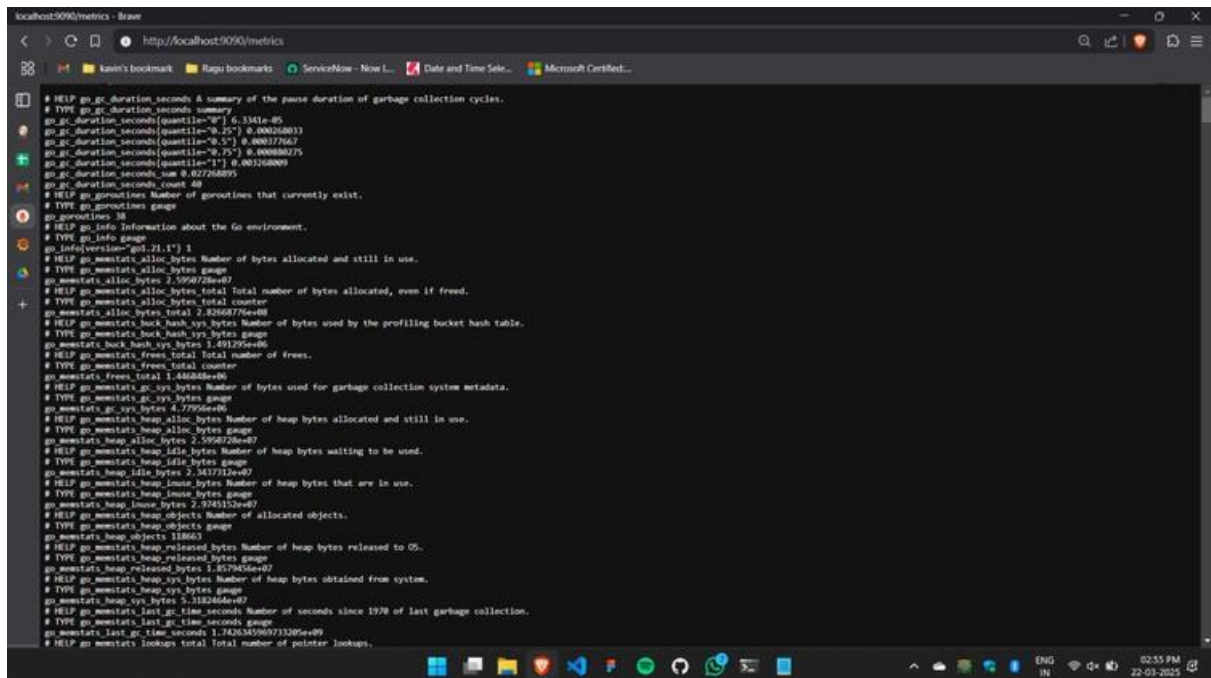
## Grafana UI:



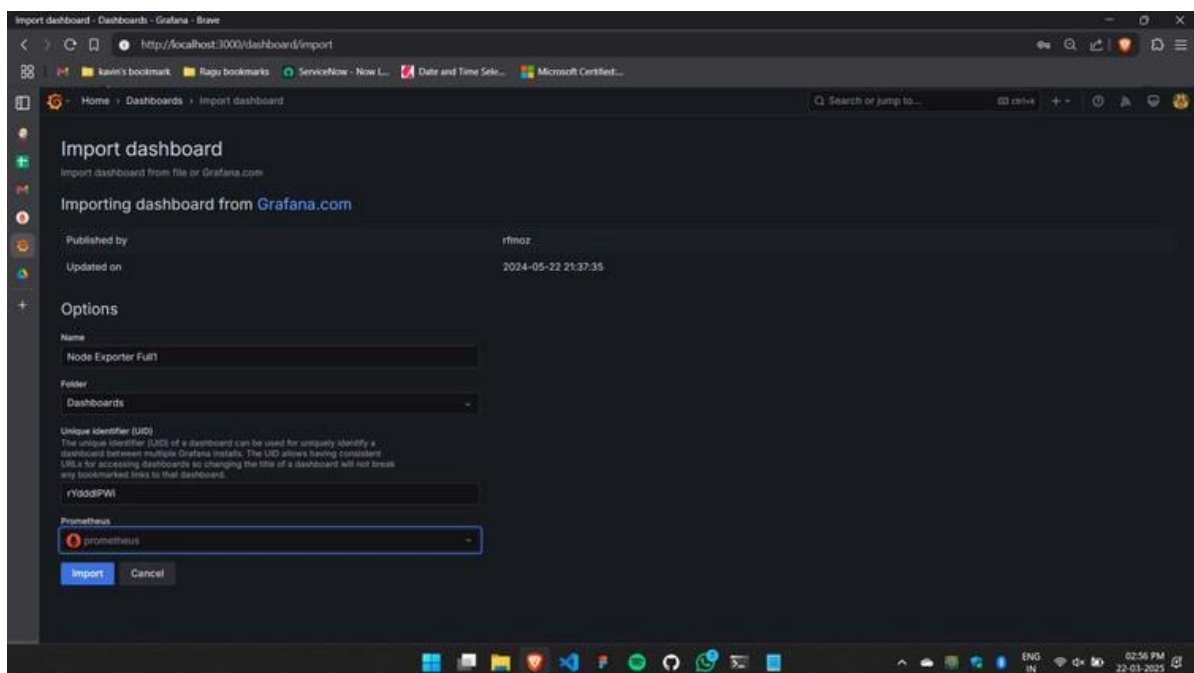
## Status in promethues:



## Metrics in Prometheus:

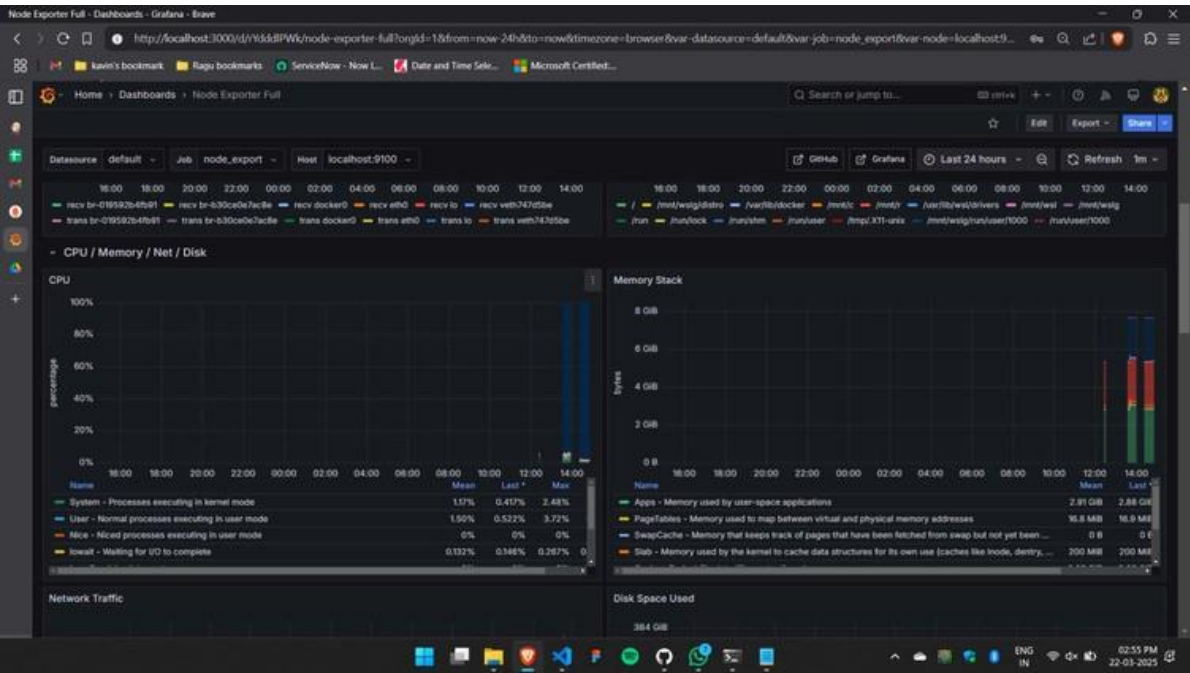
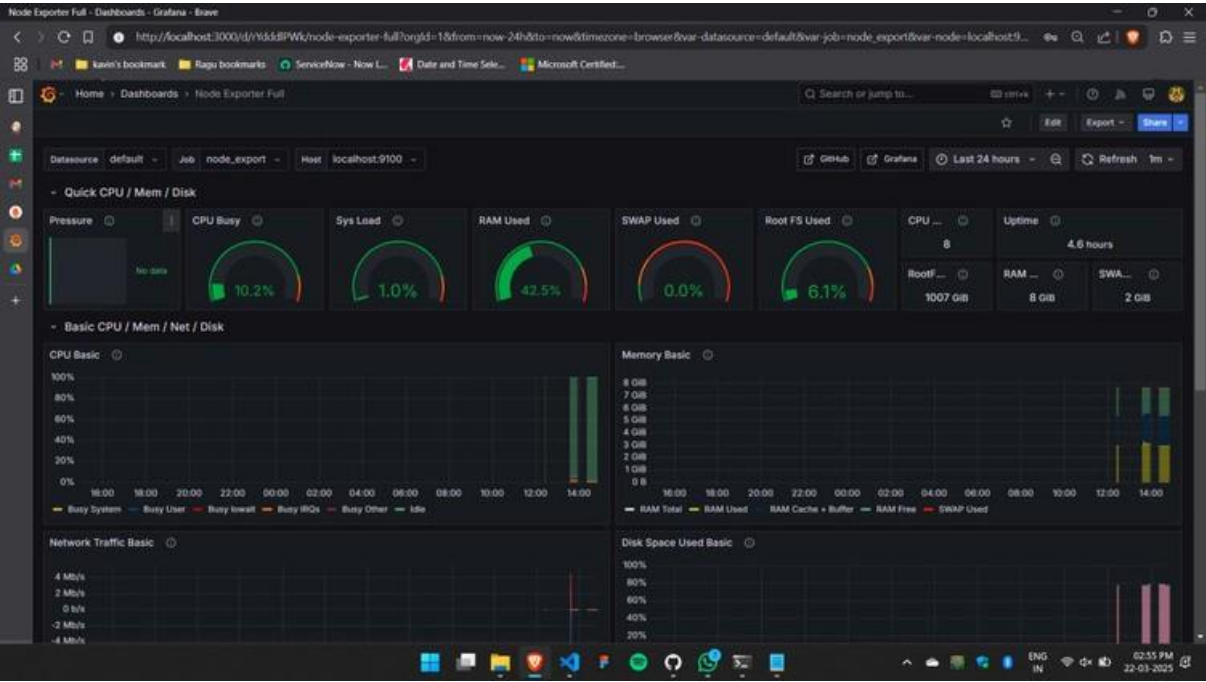


## Node Exporter:

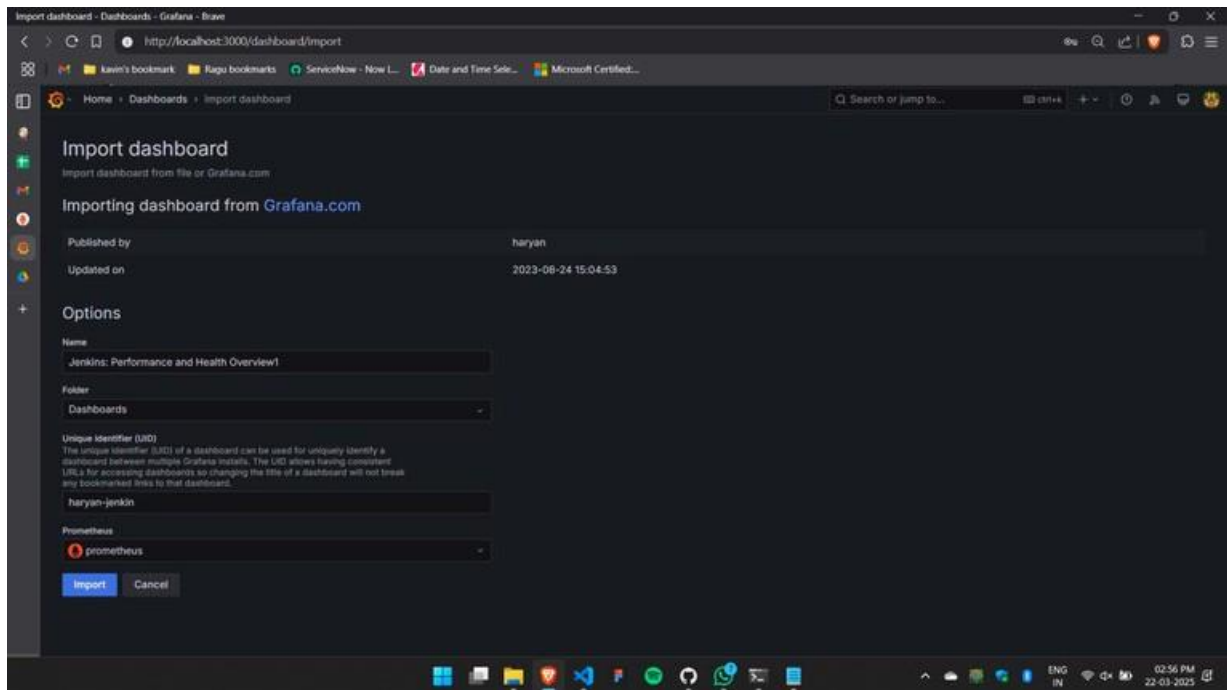




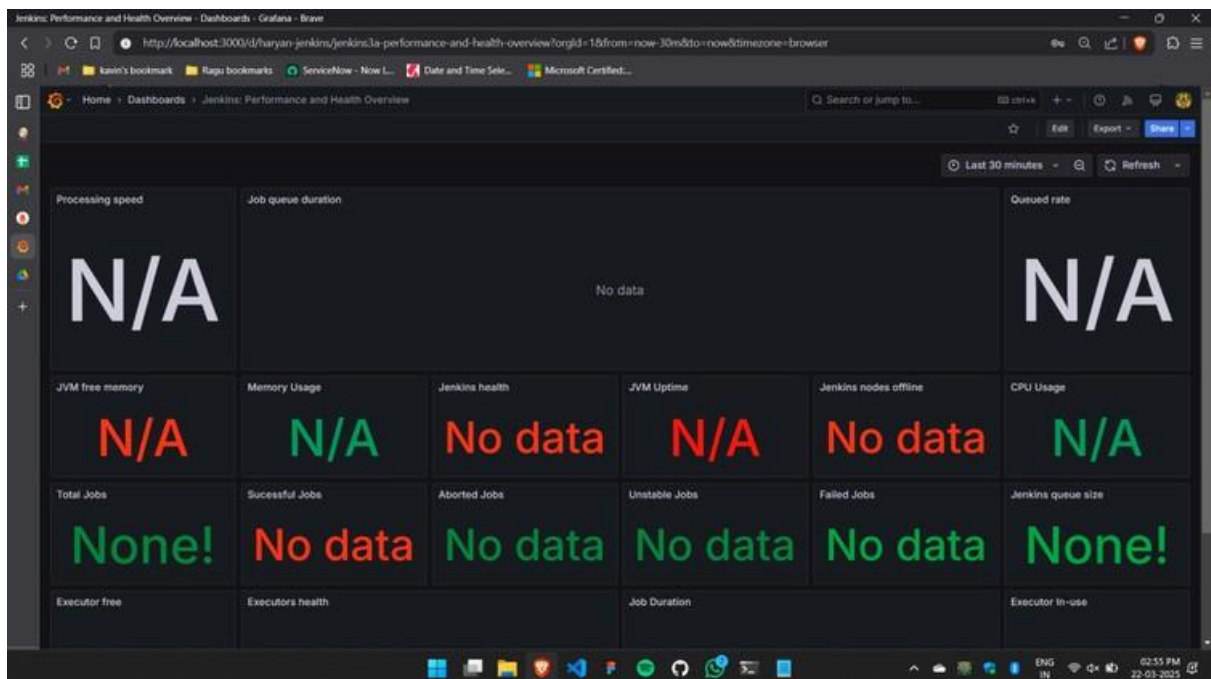
Dashboard:



## Jenkins Overview:

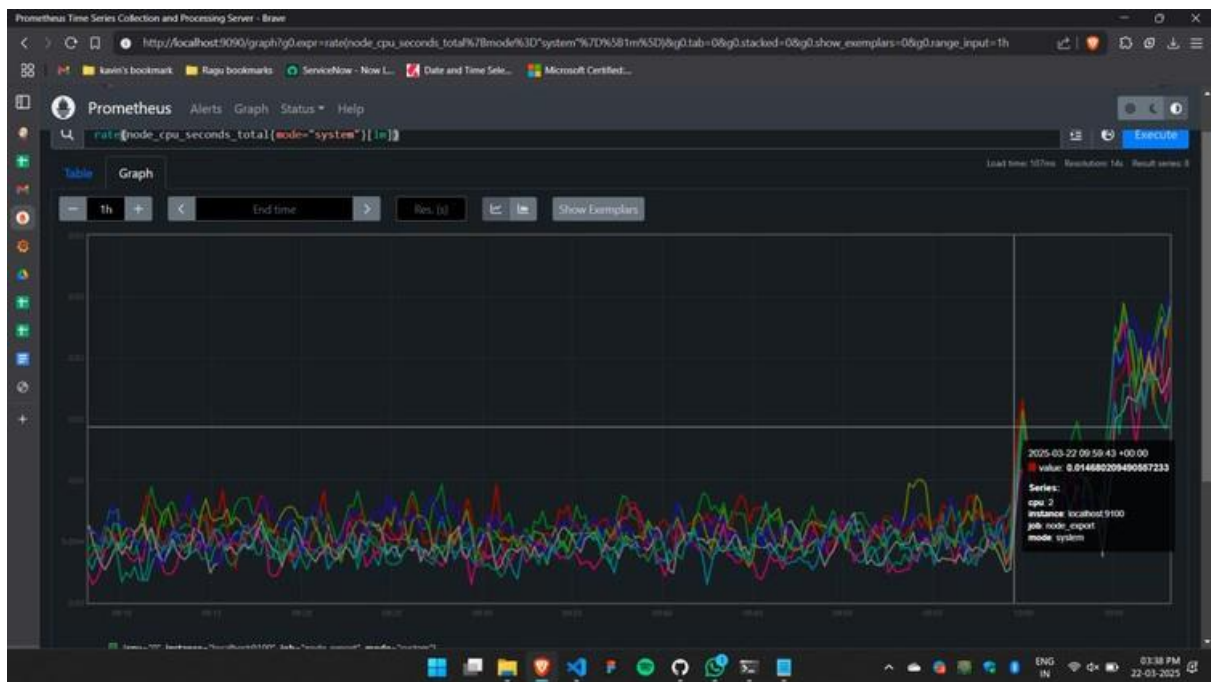


## Dashboard:

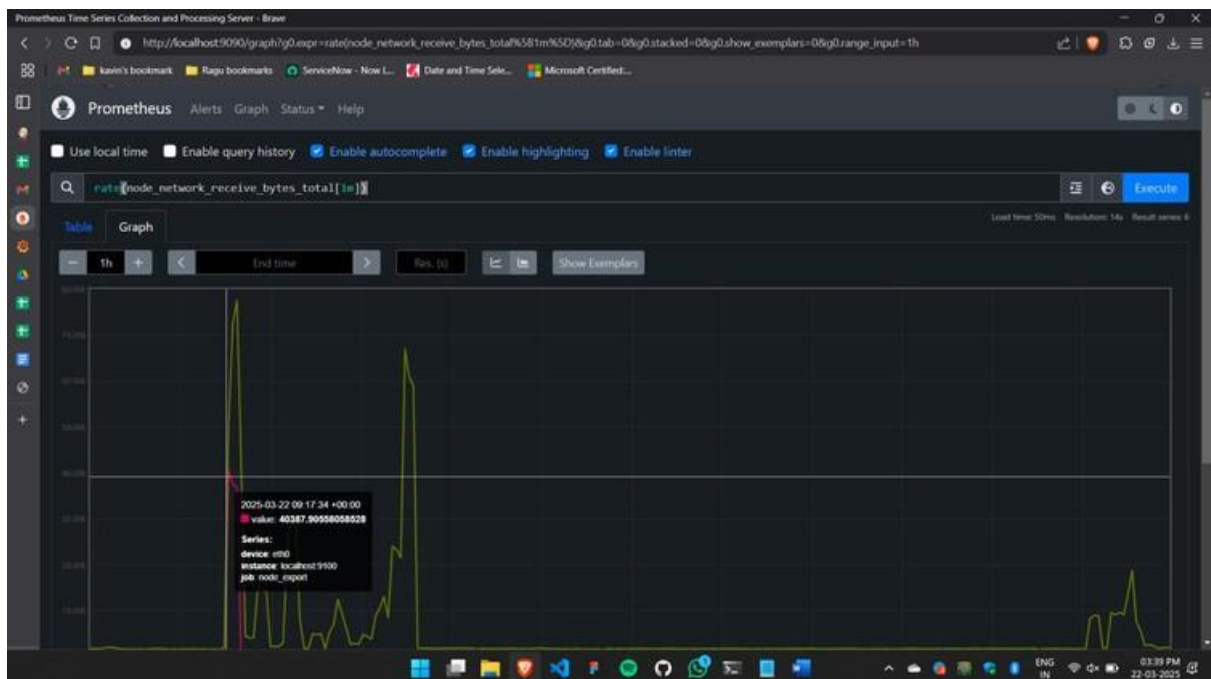


## Prometheus analysis:

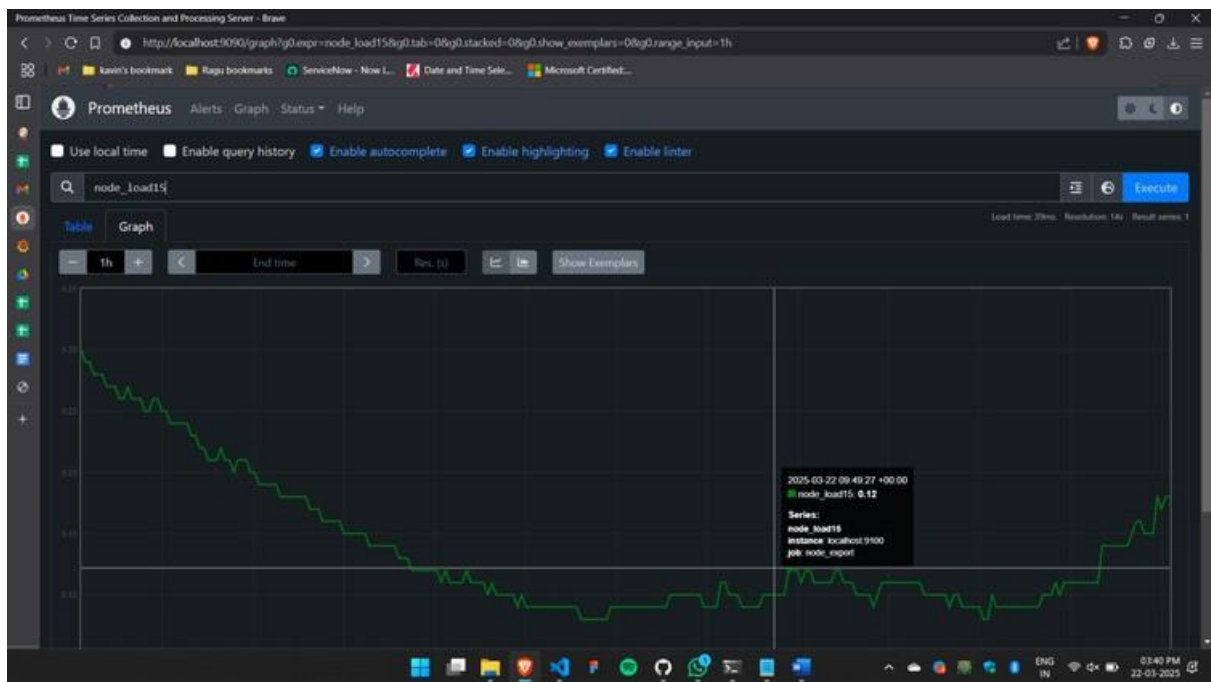
`rate(node_cpu_seconds_total{mode="system"}[1m])`



`rate(node_network_receive_bytes_total[1m])`

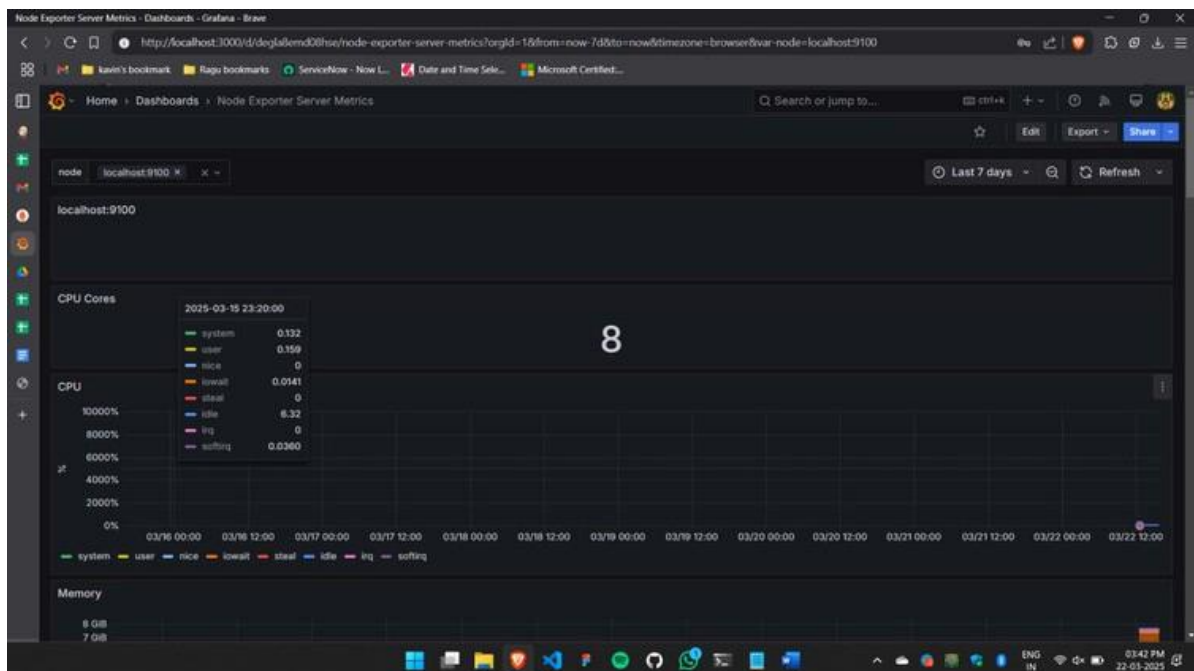


`node_load15`



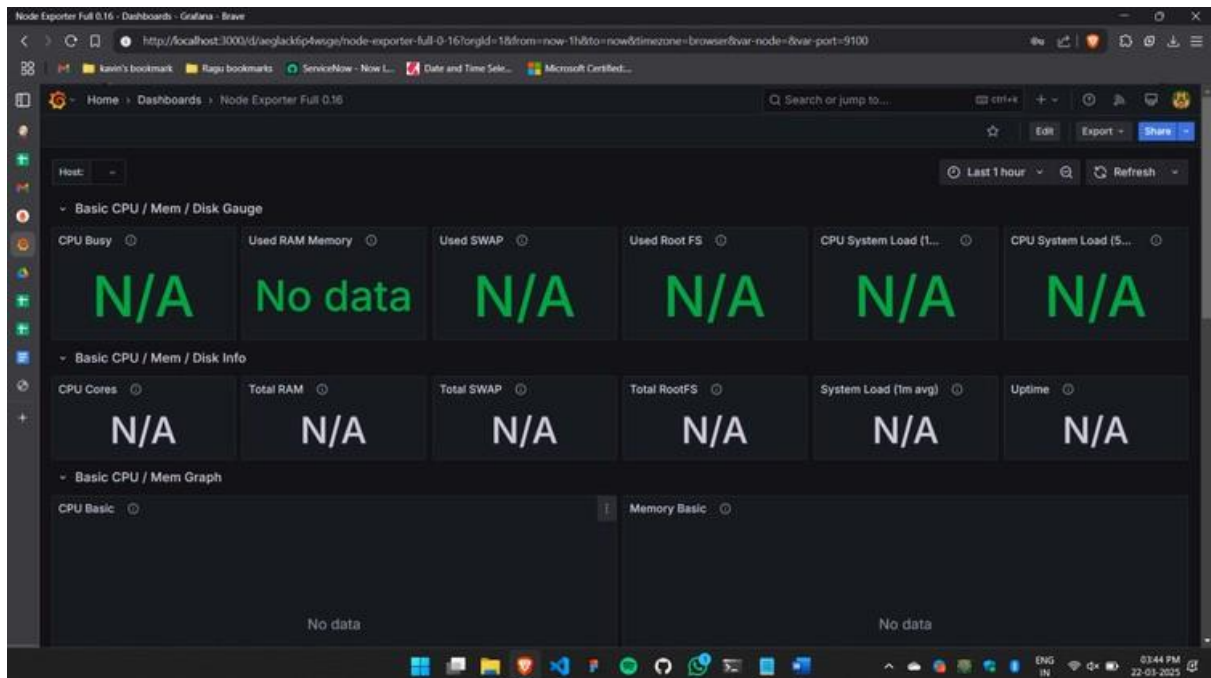
Dashboard: (405)

Node Exporter Service metrics,



Dashboard (5174):

Node Exporter Full 0.16,



Dashboard (9096):

1 Node Exporter 1.0.1

