**Monitoring Infra and Docker Container Utilization using Grafana, Prometheus, Node-exporter and Cadvisior:**

To monitoring Infra and docker container, here the below requirement is need.

Requirement:

1. Cadvisor
2. Node-exporter
3. Prometheus
4. Grafana

Note:

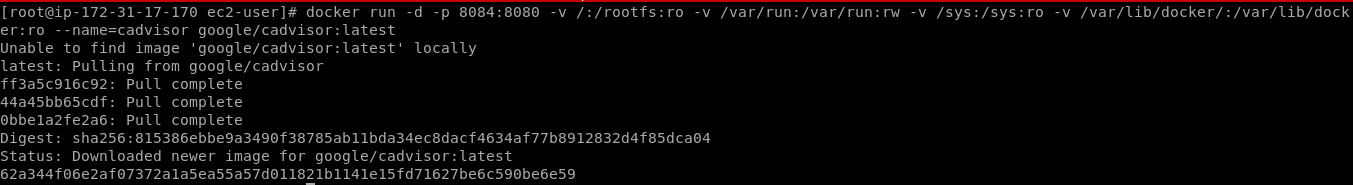
In the process we need docker engine where we can deploy above as container.

1. **Cadvisor:**

cAdvisor (Container Advisor) provides container users an understanding of the resource usage and performance characteristics of their running containers. It is a running daemon that collects, aggregates, processes, and exports information about running containers. Specifically, for each container it keeps resource isolation parameters, historical resource usage, and histograms of complete historical resource usage. This data is exported by container and machine-wide.

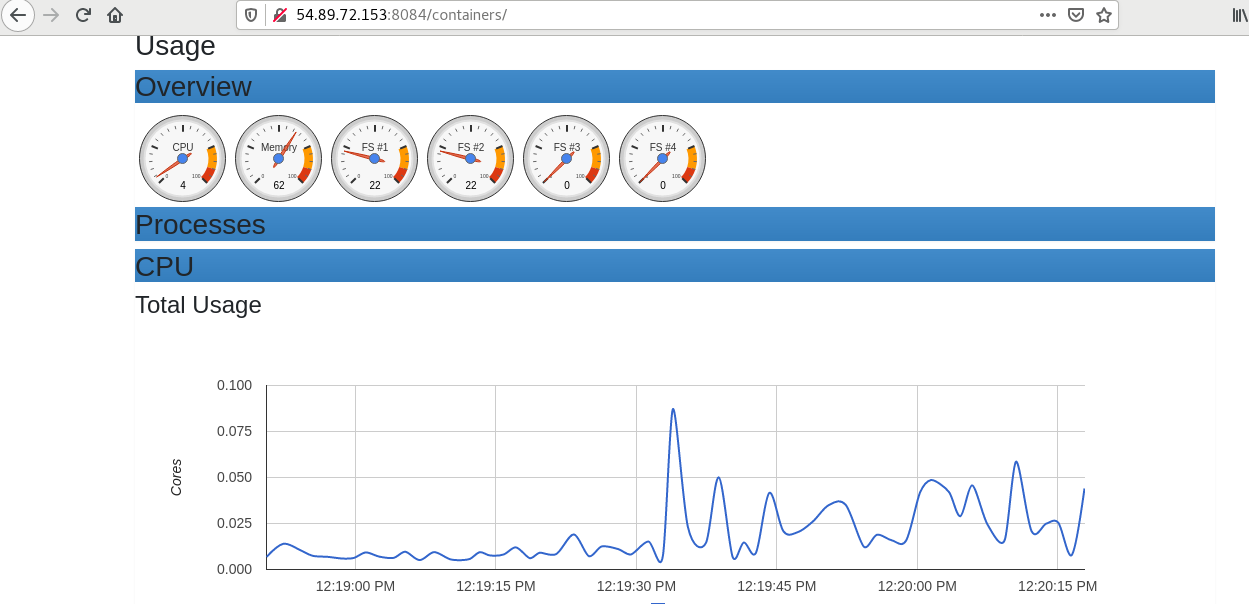
Execute the below docker command in linux server:

#docker run -d -p 8084:8080 -v /:/rootfs:ro -v /var/run:/var/run:rw -v /sys:/sys:ro -v /var/lib/docker/:/var/lib/docker:ro --name=cadvisor google/cadvisor:latest





We can access Cadvisor in browser by http://server-ip:8084



**START/STOP/RESTART Container:**

docker container start cadvisor

docker container stop cadvisor

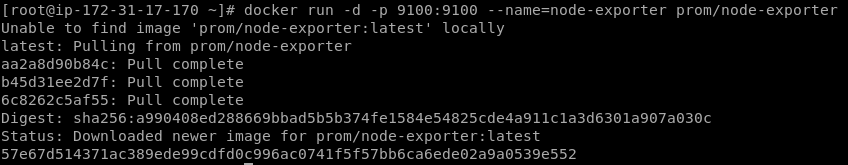
docker container restart cadvisor

2.**Node Exporter**

The node-exporter is designed to monitor the host system. It’s not recommended to deploy it as a Docker container because it requires access to the host system.

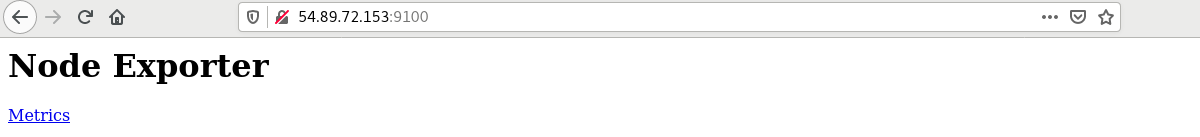
Execute the below docker command in linux server:

#docker run -d -p 9100:9100 --name=node-exporter prom/node-exporter





We can access Cadvisor in browser by [http://server-ip:9100](http://server-ip:9100/)



**START/STOP/RESTART:**

docker container start node-exporter

docker container stop node-exporter

docker container restart node-exporter

3. Prometheus

Prometheus a cloud Native computing Foundation project, is a systems and services monitoring system.

It collects metrics from configured targets at given intervals, evaluates rule expressions, displays the results, and can trigger alerts when specified conditions are observed.

Deploy Prometheus in docker:

To deploy Prometheus, we need to create configuration file for Prometheus like below,

vim /*root* *config*/prometheus.yml

global:

  scrape\_interval: 15s

  evaluation\_interval: 15s

scrape\_configs:

  - job\_name: 'prometheus'

    static\_configs:

    - targets: ['Host-IP:9090']

      labels:

        alias: 'prometheus'

  - job\_name: 'cadvisor'

    static\_configs:

    - targets: ['Host-IP:8080']

      labels:

        alias: 'cadvisor'

  - job\_name: 'node-exporter'

    static\_configs:

    - targets: ['Host-IP:9100']

      labels:

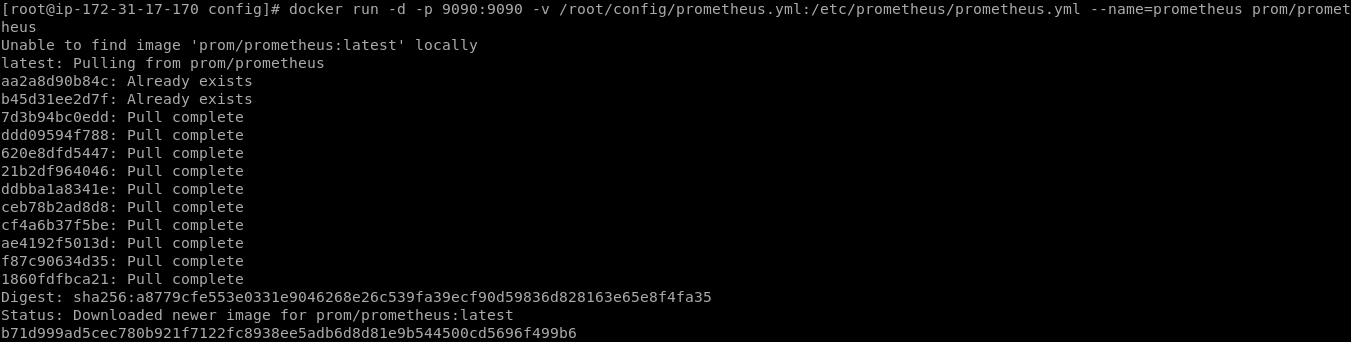
        alias: 'node-exporter'

Save the file.

Here Cadvisor, Node-exporter metrics details are given.

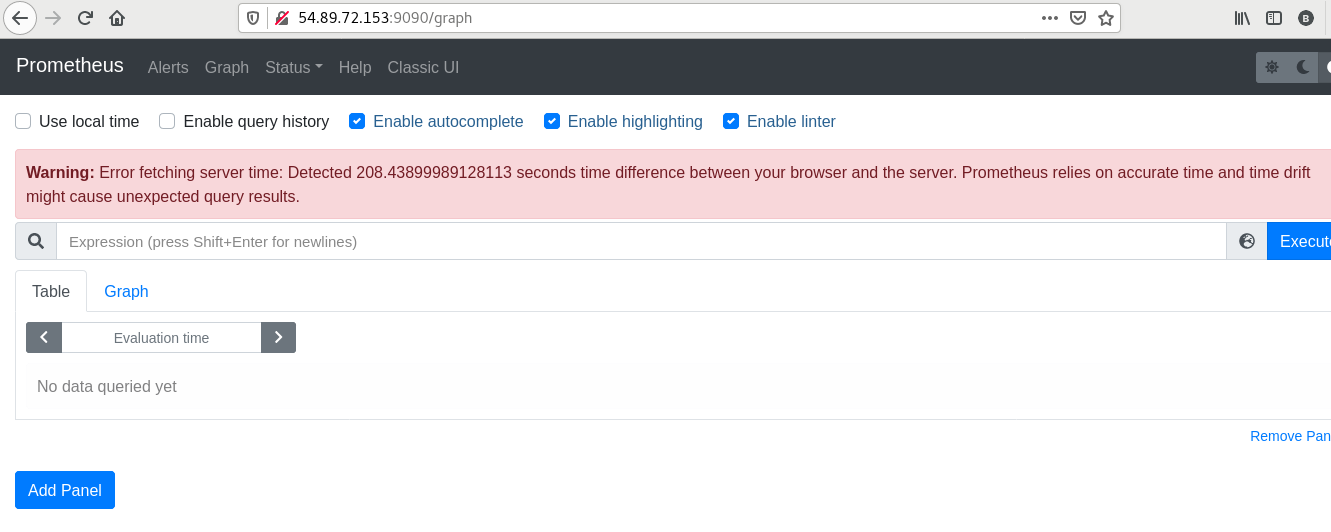
Now run the Prometheus docker command:

# docker run -d -p 9090:9090 -v /root/config/prometheus.yml:/etc/prometheus/prometheus.yml --name=prometheus prom/prometheus





We can access Cadvisor in browser by [http://server-ip:9090](http://server-ip:8084/)



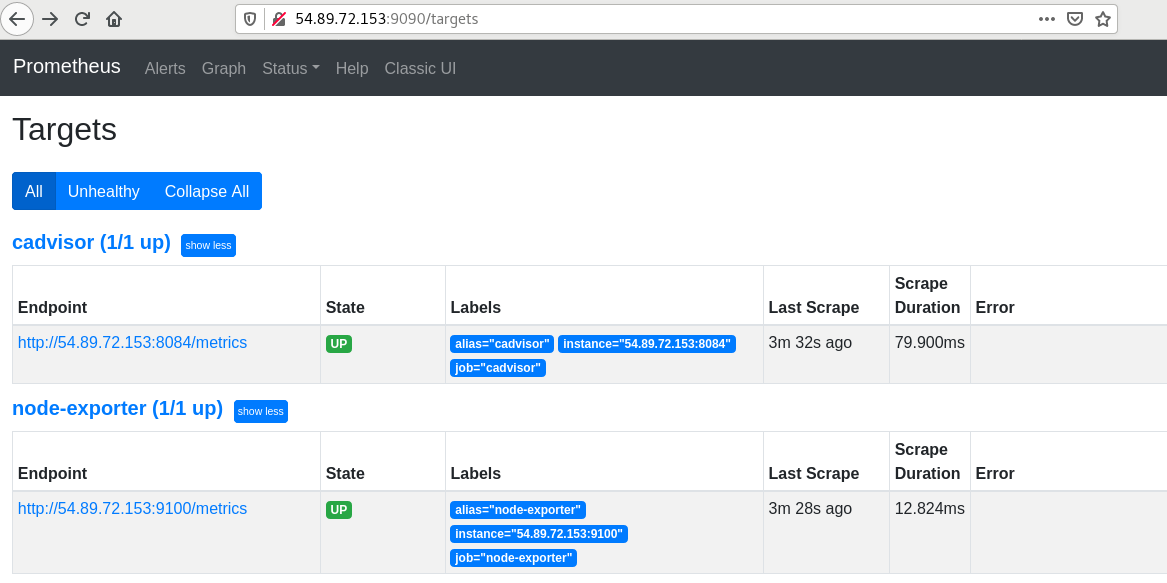
**START/RESTART/STOP Container:**

docker container start prometheus

docker container stop prometheus

docker container restart prometheus

We can check the targets are up or not in Prometheus by http://server-IP:9090/targets



4.Grafana:

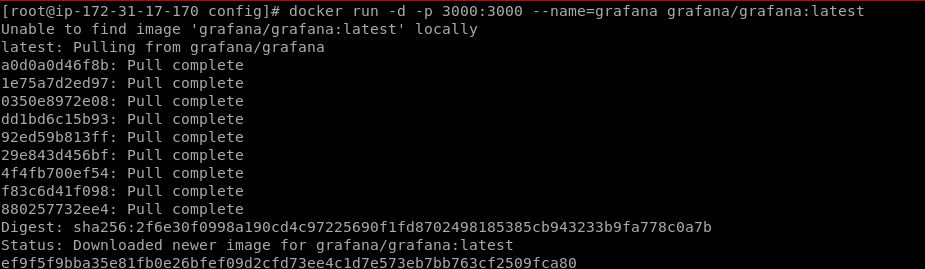
Grafana is a muliti-platform open source analytics and interactive visualization web application.

It provides charts,graphs and alerts for the web when connected to supported data sources.

Deploy Grafana in docker

Execute the below docker command in linux server.

# docker run -d -p 3000:3000 --name=grafana grafana/grafana:latest



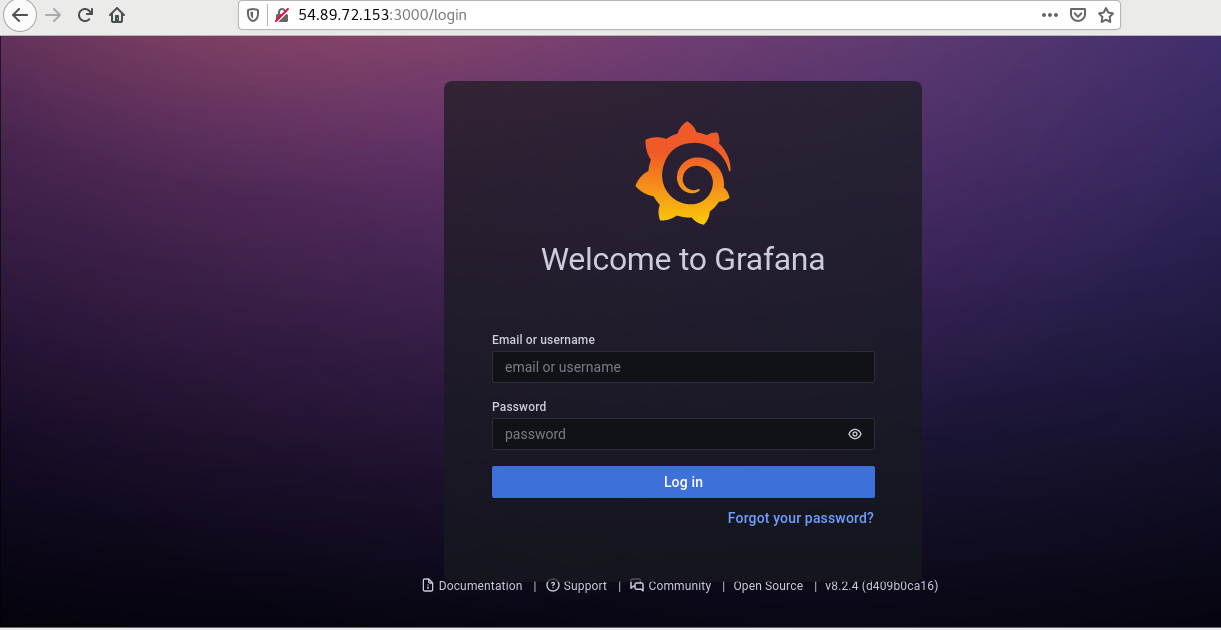
**START/STOP/RESTART Container:**

docker container start grafana

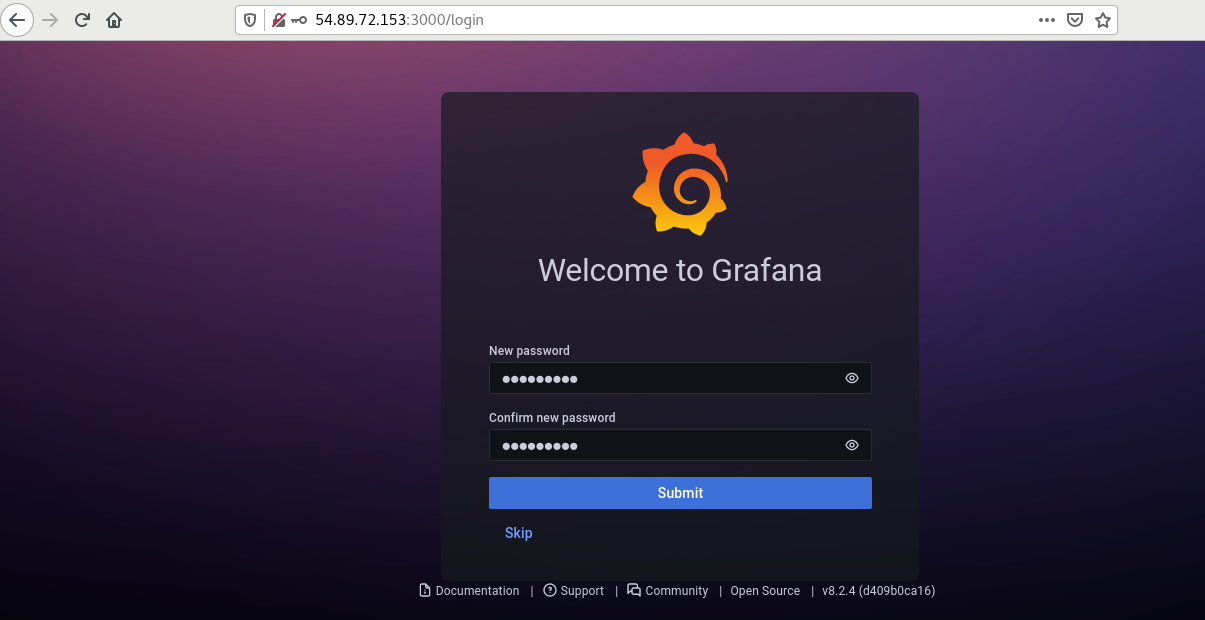
docker container stop grafana

docker container restart grafana

We can access Grafana in browser by http://server-IP:3000

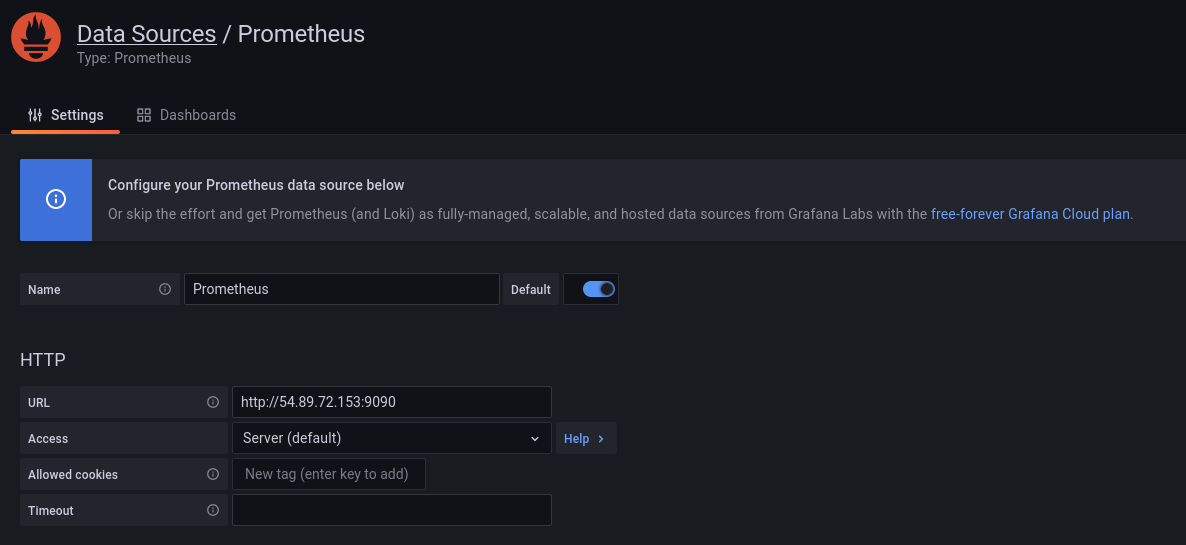


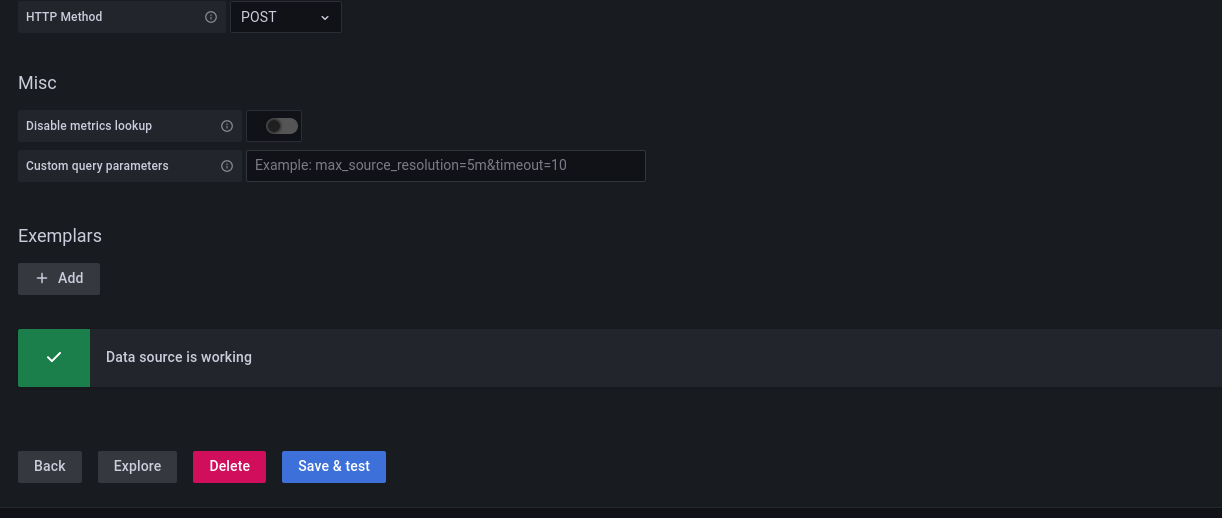
To login Grafana, the default user name and password is "admin".



Next, give new password to Grafana.

After Login to the Grafana, we need to add datasource as Prometheus in grafana.



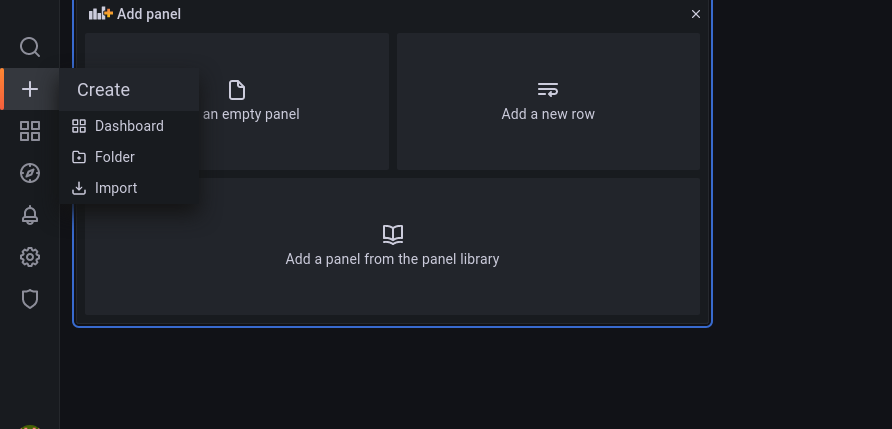


Then Save and Test and it should show like above image after click save and test.

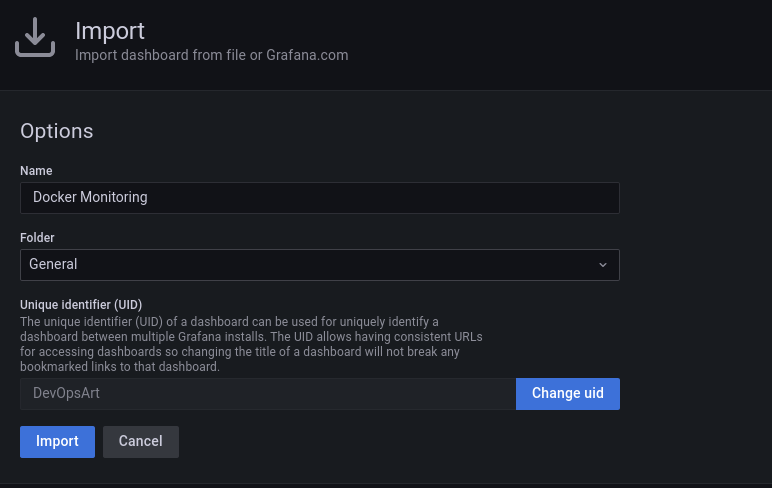
Now Create Dashboard for the monitoring the Docker container or System Utlization by the help of Prometheus metrics which has been import from cadvisior or node-expoter.

Click + sysmbol

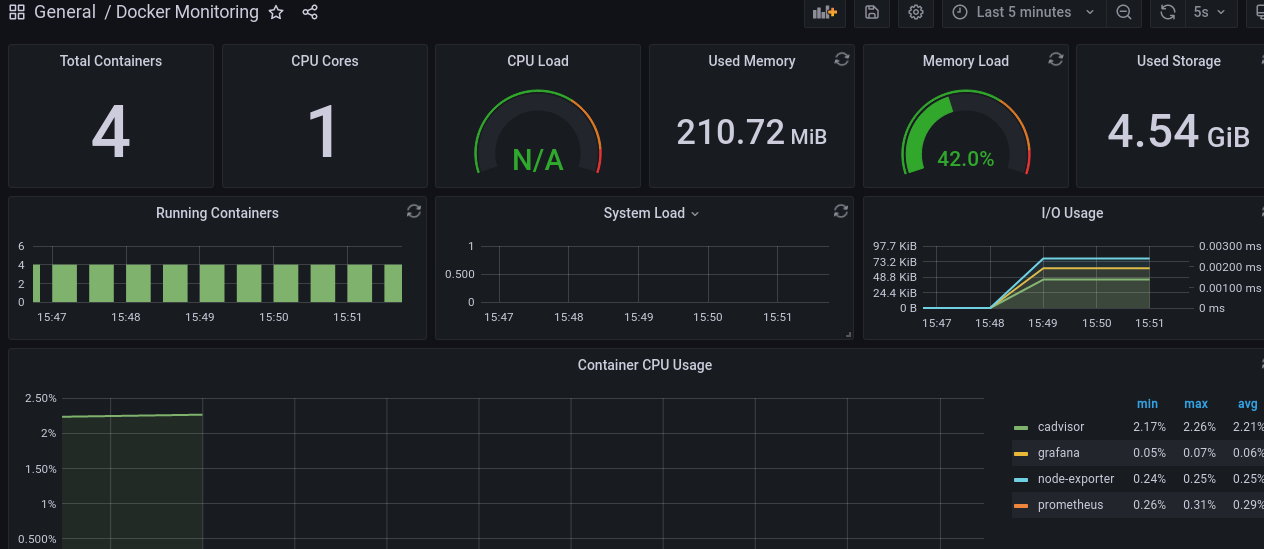
Select import and add grafana id or JSON code on panel.



After that Import the dashboard form file or grafana.



Now your Dashboard looks like below:



Here Some Grafana id example:

https://grafana.com/grafana/dashboards/1860

