Server monitoring using Prometheus & Grafana

✓ **What is Prometheus?**

- Prometheus is an open-source Linux Server Monitoring tool mainly used for metrics monitoring, event monitoring, alert management, etc.
- Prometheus has changed the way of monitoring systems and that is why it has become the Top-Level project of Cloud Native Computing Foundation (CNCF).
- Prometheus uses a powerful query language i.e., "PromQL".

❖ **Prometheus Configuration file and Components.**

- **Prometheus.yml –** It is the configuration file for Prometheus where we can do all changes regarding configuration of Prometheus.
- **Promtool** – It is command-line utility tool is used to verify the configuration of Prometheus.
- **PromQL** – Prometheus user its own query language i.e., PromQL which is very powerful querying language.
  PromQL allows the user to select and aggregate the data.

✓ **What is Grafana?**

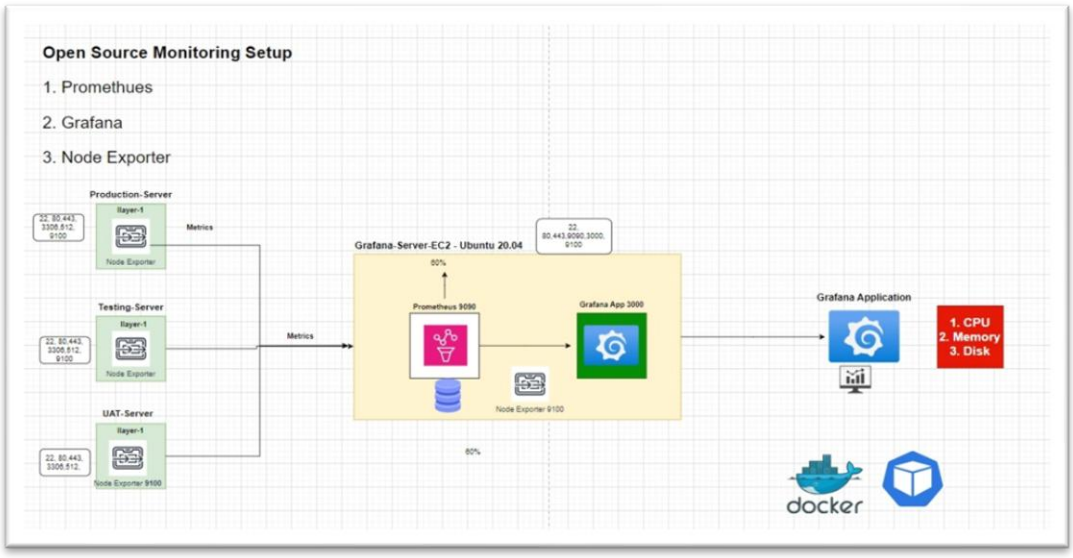
- Grafana is a free and open-source visualization tool mostly used with Prometheus to which monitor metrics.
- Grafana provides various dashboards, charts, graphs, alerts for the particular data source.
- Grafana allows us to query, visualize, explore metrics and set alerts for the data source which can be a system, server, nodes, cluster, etc.
- We can also create our own dynamic dashboard for visualization and monitoring.
- We can save the dashboard and can even share with our team members which is one of the main advantages of Grafana.

✓ **What is Node Exporter?**

- Node exporter is one of the Prometheus exporters which is used to expose servers or system OS metrics.

- With the help of Node exporter, we can expose various resources of the system like RAM, CPU utilization, Memory Utilization, disk space.

- Node exporter runs as a system service which gathers the metrics of your system and that gathered metrics is displayed with the help of Grafana visualization tool.



Prerequisites: -

- Ubuntu with 20.04 Version
- Root user account with sudo privilege.
- Prometheus system user and group.
- Sufficient storage on your system and good internet connectivity.
- Ports Required- **9090 (Prometheus), 3000 (Grafana), 9100 (Node Exporter)**

Step: - Create EC2 instance.

Step: - Add (Prometheus, Grafana, Node Exporter) Port NO. Instance attach security group.

Step: - Take SSH and configure Prometheus.

Apt update -y

- **Export the release of Prometheus**

  export RELEASE="2.2.1"

- **Creating Prometheus System Users and Directory**
  - useradd --no-create-home --shell /bin/false prometheus
  - useradd --no-create-home --shell /bin/false node_exporter
  - mkdir /etc/prometheus
  - mkdir /var/lib/Prometheus

- **Give the ownership.**
  - chown prometheus:prometheus /etc/prometheus
  - chown prometheus:prometheus /var/lib/prometheus

- **Download Prometheus Binary File**
  - cd /opt/
  - wget https://github.com/prometheus/prometheus/releases/download/v2.26.0/prometheus-2.26.0.linux-amd64.tar.gz

- **We can use sha256sum command line to generate a checksum of the Prometheus downloaded file.**
  - sha256sum prometheus-2.26.0.linux-amd64.tar.gz

- **Now we will extract the Prometheus setup file**
  - tar -xvf prometheus-2.26.0.linux-amd64.tar.gz

- **Copy Prometheus Binary files**
  - cp /opt/prometheus-2.26.0.linux-amd64/prometheus /usr/local/bin/
  - cp /opt/prometheus-2.26.0.linux-amd64/promtool /usr/local/bin/

- **Update Prometheus user ownership on Binaries**
  - chown prometheus:prometheus /usr/local/bin/prometheus
  - chown prometheus:prometheus /usr/local/bin/promtool

- **Copy Prometheus Console Libraries**
  - cp -r /opt/prometheus-2.26.0.linux-amd64/consoles /etc/prometheus
  - cp -r /opt/prometheus-2.26.0.linux-amd64/console_libraries /etc/prometheus
  - cp -r /opt/prometheus-2.26.0.linux-amd64/prometheus.yml /etc/prometheus

- **Update Prometheus ownership on Directories**
  - chown -R prometheus:prometheus /etc/prometheus/consoles
  - chown -R prometheus:prometheus /etc/prometheus/console_libraries
  - chown -R prometheus:prometheus /etc/prometheus/prometheus.yml

- **Check Prometheus Version**
  - prometheus --version
  - promtool --version

- **Prometheus configuration file**
  - cat /etc/prometheus/prometheus.yml

- **We will start it with the Prometheus user using the following command.**

```
sudo -u prometheus /usr/local/bin/prometheus \
        --config.file /etc/prometheus/prometheus.yml \
        --storage.tsdb.path /var/lib/prometheus/ \
        --web.console.templates=/etc/prometheus/consoles \
        --web.console.libraries=/etc/prometheus/console_libraries
```

- Now we will create a system service file in **/etc/systemd/system** location.
  - sudo vim /etc/systemd/system/prometheus.service

- **After creating file successfully, copy the below files**

```
[Unit]
Description=Prometheus
Wants=network-online.target
After=network-online.target
[Service]
User=Prometheus
Group=Prometheus
Type=simple
ExecStart=/usr/local/bin/prometheus \
    --config.file /etc/prometheus/prometheus.yml \
```

--storage.tsdb.path /var/lib/prometheus/ \

--web.console.templates=/etc/prometheus/consoles \

--web.console.libraries=/etc/prometheus/console_libraries

[Install]

WantedBy=multi-user.target

- systemctl daemon-reload
- systemctl start prometheus
- systemctl enable prometheus
- systemctl status Prometheus

- **enable prometheus service in firewall**
  - ufw allow 9090/tcp

- **Now Prometheus service is ready to run and we can access it from any web browser.**
  - http://server-IP-or-Hostname:9090.

Step: - Install Grafana and configure.

1. Install the prerequisite packages:
   - apt-get install -y apt-transport-https software-properties-common wget

2. Import the GPG key:
   - sudo mkdir -p /etc/apt/keyrings/
   - wget -q -O - https://apt.grafana.com/gpg.key | gpg --dearmor | sudo tee /etc/apt/keyrings/grafana.gpg > /dev/null

3. To add a repository for stable releases, run the following command:
   - echo "deb [signed-by=/etc/apt/keyrings/grafana.gpg] https://apt.grafana.com stable main" | sudo tee -a /etc/apt/sources.list.d/grafana.list

4. To add a repository for beta releases, run the following command:
   - echo "deb [signed-by=/etc/apt/keyrings/grafana.gpg] https://apt.grafana.com beta main" | sudo tee -a /etc/apt/sources.list.d/grafana.list

5. Run the following command to update the list of available packages:
   - apt-get update

6. To install Grafana OSS, run the following command:
   - apt-get install grafana

7. To install Grafana Enterprise, run the following command:
   - sudo apt-get install grafana-enterprise

8. Now start the Grafana service
   - systemctl start grafana-server

9. Now finally enable the Grafana service
   - systemctl enable grafana-server.service

10. Verify the Grafana Service Status
    - systemctl status grafana-server

- **To access Grafana Dashboard open your favourited browser.**

  http://43.205.198.235/:3000

- **Here you can see Login page of Grafana now you will have to login with below Grafana default Username and Password.**

  Username – **admin**

  Password – **admin**

Step: - Install Node Exporter on Ubuntu.

- wget https://github.com/prometheus/node_exporter/releases/download/v1.2.0/node_exporter-1.2.0.linux-amd64.tar.gz

- **Unzip the downloaded the file**

  tar xvzf node_exporter-1.2.0.linux-amd64.tar.gz

- **Go to that file and move this file to your /usr/local/bin directory using below command**

    - cd node_exporter-1.2.0.linux-amd64
    - cp node_exporter /usr/local/bin

- **Creating Node Exporter Systemd service**

    - cd /lib/systemd/system
    - vim node_exporter.service

- **Paste the below content in your service file**

```
[Unit]
Description=Node Exporter
Wants=network-online.target
After=network-online.target
[Service]
Type=simple
User=node_exporter
Group=node_exporter
ExecStart=/usr/local/bin/node_exporter \
— collector.mountstats \
— collector.logind \
— collector.processes \
— collector.ntp \
— collector.systemd \
— collector.tcpstat \
— collector.wifi
Restart=always
RestartSec=10s
[Install]
WantedBy=multi-user.target
```

- **Now lets start and enable the node_exporter service using below commands**
    - systemctl daemon-reload
    - systemctl enable node_exporter
    - systemctl start node_exporter
    - systemctl status node_exporter

- **Configure the Node Exporter as a Prometheus target**
    - cd /etc/prometheus
    - vim prometheus.yml
    - inside Prometheus.yml in this file add 3000 port inside then try to hit the ip

Final Result :