# Prometheus & Grafana Containerized Deployment

## Node Exporter:

Install node exporter on the servers you’d like to monitor.

* Create a shellscript file and paste the following below to install the node exporter

#!/bin/bash

cd /tmp

wget https://github.com/prometheus/node\_exporter/releases/download/v1.5.0/node\_exporter-1.5.0.linux-amd64.tar.gz

tar -xf node\_exporter-1.5.0.linux-amd64.tar.gz

sudo mv node\_exporter-1.5.0.linux-amd64/node\_exporter /usr/local/bin

#rm -r node\_exporter-1.5.0.linux-amd64\*

sudo useradd -rs /bin/false node\_exporter

cat > /etc/systemd/system/node\_exporter.service << "EOF"

[Unit]

Description=Node Exporter

After=network.target

[Service]

User=node\_exporter

Group=node\_exporter

Type=simple

ExecStart=/usr/local/bin/node\_exporter --web.listen-address=:1784

[Install]

WantedBy=multi-user.target

EOF

sudo systemctl daemon-reload

sudo systemctl enable node\_exporter

sudo systemctl start node\_exporter

## Install Docker in your server

Create a shellscript file to install docker

#!/bin/bash

sudo apt-get update

sudo apt-get install ca-certificates curl

sudo install -m 0755 -d /etc/apt/keyrings

sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o /etc/apt/keyrings/docker.asc

sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:

echo \

"deb [arch=$(dpkg --print-architecture) signed-by=/etc/apt/keyrings/docker.asc] https://download.docker.com/linux/ubuntu \

$(. /etc/os-release && echo "$VERSION\_CODENAME") stable" | \

sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update

sudo apt-get install docker-ce docker-ce-cli containerd.io docker-buildx-plugin docker-compose-plugin

## Create an IAM role to gather required info

Create an IAM role and give it read-only permissions for EC2, RDS, Lambda, Cloudwatch

## Create a docker-compose.yaml file

First, create a directory to store all the files

* mkdir prometheus-grafana

create a docker-compose.yaml file and paste the below contents:

version: '3.8'

networks:

monitoring:

driver: bridge

volumes:

prometheus-data:

driver: local

grafana-data:

driver: local

services:

prometheus: # Prometheus Configuration

image: prom/prometheus:v2.37.9

container\_name: prometheus

ports:

- 1783:9090

command:

- '--config.file=/etc/prometheus/prometheus.yaml'

volumes:

- prometheus-grafana/config/:/etc/prometheus/

- ./data:/prometheus/data/

restart: unless-stopped

grafana: # Grafana

image: grafana/grafana:latest

container\_name: grafana

ports:

- '3000:3000'

volumes:

- prometheus-grafana/grafana-data:/var/lib/grafana

environment:

GF\_RENDERING\_SERVER\_URL: http://renderer:8081/render

GF\_RENDERING\_CALLBACK\_URL: http://grafana:3000/

GF\_LOG\_FILTERS: rendering:debug

renderer:

image: grafana/grafana-image-renderer:latest

ports:

- 8081

restart: unless-stopped

asciidoctor: #ASCIIdoc for rendering

image: asciidoctor/docker-asciidoctor

container\_name: asciidoctor

ports:

- "8815:8815"

hostname: asciidoctor

volumes:

- prometheus-grafana/reportgen:/documents

command:

sh /documents/startup.sh

restart: unless-stopped

alertmanager: #AlertManager

image: prom/alertmanager:latest

container\_name: alertmanager

ports:

- "9093:9093"

volumes:

- prometheus-grafana/alertmanager:/etc/alertmanager

restart: unless-stopped

## Creating config files for the container:

Create the below directories:

* mkdir prometheus-grafana/config
* mkdir prometheus-grafana/grafana-data
* mkdir prometheus-grafana/reportgen
* mkdir prometheus-grafana/alertmanager

### Goto the directory: prometheus-grafana/config/ and create a prometheus.yml file and paste the below configuration:

global:

scrape\_interval: 1m

evaluation\_interval: 1m

scrape\_configs:

- job\_name: "node-linux"

ec2\_sd\_configs: &ec2config

- region: "ap-south-1"

role\_arn: "<role arn>"

access\_key: ""

secret\_key: ""

relabel\_configs:

- source\_labels: [\_\_meta\_ec2\_tag\_OS]

regex: linux

action: keep

- source\_labels: [\_\_meta\_ec2\_private\_ip]

regex: '(.\*)'

replacement: '${1}:1784'

target\_label: \_\_address\_\_

- source\_labels: [\_\_meta\_ec2\_tag\_Name]

target\_label: instance\_name

replacement: "youremailid@email.com"

- job\_name: "node-windows"

ec2\_sd\_configs: \*ec2config

relabel\_configs:

- source\_labels: [\_\_meta\_ec2\_tag\_OS]

regex: windows

action: keep

- source\_labels: [\_\_meta\_ec2\_private\_ip]

regex: '(.\*)'

replacement: '${1}:9182'

target\_label: \_\_address\_\_

- source\_labels: [\_\_meta\_ec2\_tag\_Name]

target\_label: instance\_name

- target\_label: "cust\_email\_id"

replacement: "youremail@email.com"

- job\_name: 'blackbox-https-url'

metrics\_path: /probe

params:

module: [http\_2xx]

static\_configs:

- targets:

# - https://minfytech.com

relabel\_configs:

- source\_labels: [\_\_address\_\_]

target\_label: \_\_param\_target

- source\_labels: [\_\_param\_target]

target\_label: instance

- target\_label: \_\_address\_\_

replacement: localhost:9115

rule\_files:

- "/etc/prometheus/EC2-Alerts.yml"

- "/etc/prometheus/URL-Alerts.yml"

alerting:

alertmanagers:

- static\_configs:

- targets: ["publicip:9093"]

### Goto the directory: prometheus-grafana/alertmanager and create Ec2-alert.yaml, RDS-alert.yaml and URL-alert.yaml

### EC2.yaml >>

groups:

- name: disk\_usage\_alerts

rules:

- alert: HostDiskUtilizationTrouble

expr: (node\_filesystem\_size\_bytes - node\_filesystem\_free\_bytes) / node\_filesystem\_size\_bytes \* 100 >= 75

for: 5m

labels:

severity: trouble

annotations:

description: "Disk utilization is over 75%. The Monitor {{ $labels.instance\_name }} is in a TROUBLE state. The possible reason for this incident is Disk Utilization exceeds 75% filesystem {{ $labels.mountpoint }}."

VALUE: '{{ $value }}'

LABELS: '{{ $labels }}'

summary: "Host is in trouble due to high disk utilization (instance {{ $labels.instance\_name }}, filesystem {{ $labels.mountpoint }})"

- alert: HostDiskUtilizationCritical

expr: (node\_filesystem\_size\_bytes - node\_filesystem\_free\_bytes) / node\_filesystem\_size\_bytes \* 100 >= 90

for: 5m

labels:

severity: critical

annotations:

description: "Disk utilization is over 90%. The Monitor {{ $labels.instance\_name }} is in a CRITICAL state. The possible reason for this incident is Disk Utilization exceeds 90% filesystem {{ $labels.mountpoint }}."

VALUE: '{{ $value }}'

LABELS: '{{ $labels }}'

summary: "Host is in a critical condition due to extremely high disk utilization (instance {{ $labels.instance\_name }}, filesystem {{ $labels.mountpoint }})"

- name: disk\_partition\_usage\_alerts

rules:

- alert: HostDiskPartitionUsageCritical

expr: (node\_filesystem\_size\_bytes - node\_filesystem\_avail\_bytes) / node\_filesystem\_size\_bytes \* 100 >= 90

for: 5m

labels:

severity: critical

annotations:

description: "Disk partition usage is over 90%. The Monitor {{ $labels.instance\_name }} is in a CRITICAL state. The possible reason for this incident is Disk Partition Usage exceeds 90% partition {{ $labels.mountpoint }}."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host disk partition is in a critical condition (instance {{ $labels.instance\_name }}, partition {{ $labels.mountpoint }})"

- alert: HostDiskPartitionUsageTrouble

expr: (node\_filesystem\_size\_bytes - node\_filesystem\_avail\_bytes) / node\_filesystem\_size\_bytes \* 100 >= 75

for: 5m

labels:

severity: trouble

annotations:

description: "Disk partition usage is over 75%. The Monitor {{ $labels.instance\_name }} is in a TROUBLE state. The possible reason for this incident is DiskPartition Usage exceeds 75% partition {{ $labels.mountpoint }}."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host disk partition is in trouble (instance {{ $labels.instance\_name }}, partition {{ $labels.mountpoint }})"

- name: diskio\_alerts

rules:

- alert: DiskIOHighCritical

expr: rate(node\_disk\_written\_bytes\_total{device=~"^(sd.\*|xv.\*|nvme.\*)$"}[1m]) > 10000000

for: 5m

labels:

severity: critical

annotations:

description: "Disk I/O ({{ printf \"%.2f\" $value }} bytes/sec) on {{ $labels.instance\_name }} is critically high, exceeding the threshold of 10MB/s. Immediate investigation required."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "High disk I/O detected on {{ $labels.instance\_name }}"

- name: server\_alerts

rules:

- alert: ServerDown

expr: up == 0

for: 5m

labels:

severity: critical

annotations:

description: "Server {{ $labels.instance\_name }} is down. The Monitor {{ $labels.instance\_name }} is in a CRITICAL state. The possible reason for this incident is the server is not responding."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Server {{ $labels.instance\_name }} is down"

- name: memory\_alerts

rules:

- alert: HostMemoryCritical

expr: node\_memory\_MemAvailable\_bytes / node\_memory\_MemTotal\_bytes \* 100 < 10

for: 5m

labels:

severity: critical

annotations:

description: "Node memory is critically low (less than 10% available). The Monitor {{ $labels.instance\_name }} is in a CRITICAL state. The possible reason for this incident is Memory Usage exceeds 90%."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host is in a critical condition due to extremely low memory (instance {{ $labels.instance\_name }})"

- alert: HostMemoryTrouble

expr: node\_memory\_MemAvailable\_bytes / node\_memory\_MemTotal\_bytes \* 100 < 25

for: 5m

labels:

severity: trouble

annotations:

description: "Node memory is running low (less than 25% available). The Monitor {{ $labels.instance\_name }} is in a TROUBLE state. The possible reason for this incident is Memory Usage exceeds 75%."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host is in trouble due to low memory (instance {{ $labels.instance\_name }})"

- name: swapmemory\_alerts

rules:

- alert: HostHighSwapUsageCritical

expr: (node\_memory\_SwapTotal\_bytes - node\_memory\_SwapFree\_bytes) / node\_memory\_SwapTotal\_bytes > 0.90

for: 5m

labels:

severity: critical

annotations:

description: "Swap memory usage is above 90%. The Monitor {{ $labels.instance\_name }} is in a CRITICAL state. The possible reason for this incident is Swap Memory Usage exceeds 90%."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host is in a critical condition due to extremely high swap memory usage (instance {{ $labels.instance\_name }})"

- alert: HostHighSwapUsageTrouble

expr: (node\_memory\_SwapTotal\_bytes - node\_memory\_SwapFree\_bytes) / node\_memory\_SwapTotal\_bytes > 0.75

for: 5m

labels:

severity: trouble

annotations:

description: "Swap memory usage is above 75%. The Monitor {{ $labels.instance\_name }} is in a TROUBLE state. The possible reason for this incident is Swap Memory Usage exceeds 75%."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host might be in trouble due to high swap memory usage (instance {{ $labels.instance\_name }})"

- name: cpu\_alerts

rules:

- alert: HostHighCpuLoadCritical

expr: rate(node\_cpu\_seconds\_total{mode!="idle"}[2m]) > 0.90

for: 5m

labels:

severity: critical

annotations:

description: "CPU load is > 90. The Monitor {{ $labels.instance\_name }} is in a CRITICAL state. The possible reason for this incident is CPU load exceeds 90."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host is in a critical condition due to extremely high CPU load (instance {{ $labels.instance\_name }})"

- alert: HostHighCpuLoadTrouble

expr: rate(node\_cpu\_seconds\_total{mode!="idle"}[2m]) > 0.75

for: 5m

labels:

severity: trouble

annotations:

description: "CPU load is > 75%. The Monitor {{ $labels.instance\_name }} is in a TROUBLE state. The possible reason for this incident is CPU load exceeds 75%."

VALUE: "{{ $value }}"

LABELS: "{{ $labels }}"

summary: "Host is in trouble due to high CPU load (instance {{ $labels.instance\_name }})"

### RDS-alert.yaml >>

groups:

- name: RDSInstanceBurstBalanceAlerts

rules:

- alert: RDSInstanceBurstBalanceCritical

expr: avg\_over\_time(aws\_rds\_burst\_balance\_average[15m]) > 90

for: 10s

labels:

severity: critical

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Burst Balance is > 90% over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} might be underutilizing its I/O capacitycritically.'

summary: 'Critical: RDS instance {{ $labels.dbinstance\_identifier }} has very high burst balance'

- alert: RDSInstanceBurstBalanceTrouble

expr: avg\_over\_time(aws\_rds\_burst\_balance\_average[15m]) > 75

for: 10s

labels:

severity: trouble

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Burst Balance is > 75% over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} might be underutilizing its I/O capacity.'

summary: 'Warning: RDS instance {{ $labels.dbinstance\_identifier }} has high burst balance'

- alert: RDSInstanceBurstBalanceResolved

expr: avg\_over\_time(aws\_rds\_burst\_balance\_average[15m]) <= 75

for: 10s

labels:

severity: resolved

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Burst Balance is at or below 75% over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} is utilizing its I/O capacity efficiently.'

summary: 'Resolved: RDS instance {{ $labels.dbinstance\_identifier }} has an acceptable burst balance'

- name: RDSInstanceFreeableMemoryAlerts

rules:

- alert: RDSInstanceFreeableMemoryTrouble

expr: avg\_over\_time(aws\_rds\_freeable\_memory\_average[15m]) > 500

for: 10s

labels:

severity: trouble

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Freeable memory is above 500 bytes over the past 15 minutes, indicating potential memory underutilization or leak. The RDS instance {{ $labels.dbinstance\_identifier }} might need attention.'

summary: 'Warning: RDS instance {{ $labels.dbinstance\_identifier }} has high freeable memory'

- alert: RDSInstanceFreeableMemoryCritical

expr: avg\_over\_time(aws\_rds\_freeable\_memory\_average[15m]) > 3741824

for: 10s

labels:

severity: critical

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Freeable memory is above 3.74 MB over the past 10 minutes, indicating serious memory underutilization or leak. Immediate attention required for RDS instance {{ $labels.dbinstance\_identifier }}.'

summary: 'Critical: RDS instance {{ $labels.dbinstance\_identifier }} has very high freeable memory'

- alert: RDSInstanceFreeableMemoryResolved

expr: avg\_over\_time(aws\_rds\_freeable\_memory\_average[15m]) <= 500

for: 10s

labels:

severity: resolved

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Freeable memory is at or below 500 bytes over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} is utilizing its memory efficiently.'

summary: 'Resolved: RDS instance {{ $labels.dbinstance\_identifier }} has an acceptable level of freeable memory'

- name: RDSInstanceCPUAlerts

rules:

- alert: RDSInstanceCPUCritical

expr: avg\_over\_time(aws\_rds\_cpuutilization\_average[15m]) > 90

for: 10s

labels:

severity: critical

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'CPU utilization is above 90% over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} might be experiencing high load.'

summary: 'Critical: RDS instance {{ $labels.dbinstance\_identifier }} has high CPU usage'

- alert: RDSInstanceCPUTrouble

expr: avg\_over\_time(aws\_rds\_cpuutilization\_average[15m]) > 75

for: 10s

labels:

severity: trouble

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'CPU utilization is above 75% over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} is under significant load.'

summary: 'Warning: RDS instance {{ $labels.dbinstance\_identifier }} has elevated CPU usage'

- alert: RDSInstanceCPUResolved

expr: avg\_over\_time(aws\_rds\_cpuutilization\_average[15m]) <= 75

for: 10s

labels:

severity: resolved

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'CPU utilization is at or below 75% over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} is operating within normal parameters.'

summary: 'Resolved: RDS instance {{ $labels.dbinstance\_identifier }} has normal CPU usage'

- name: RDSInstanceDBConnectionsAlerts

rules:

- alert: RDSInstanceDBConnectionsCritical

expr: avg\_over\_time(aws\_rds\_database\_connections\_average[15m]) > 3000

for: 10s

labels:

severity: critical

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Database connections are above 3000 over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} might be experiencing extremely high load.'

summary: 'Critical: RDS instance {{ $labels.dbinstance\_identifier }} has very high database connections'

- alert: RDSInstanceDBConnectionsTrouble

expr: avg\_over\_time(aws\_rds\_database\_connections\_average[15m]) > 1000 and avg\_over\_time(aws\_rds\_database\_connections\_average[15m]) <= 3000

for: 10s

labels:

severity: trouble

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Database connections are between 1000 and 3000 over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} is under significant load.'

summary: 'Warning: RDS instance {{ $labels.dbinstance\_identifier }} has high database connections'

- alert: RDSInstanceDBConnectionsResolved

expr: avg\_over\_time(aws\_rds\_database\_connections\_average[15m]) <= 1000

for: 10s

labels:

severity: resolved

annotations:

LABELS: '{{ $labels }}'

VALUE: '{{ $value }}'

description: 'Database connections are at or below 1000 over the past 15 minutes. The RDS instance {{ $labels.dbinstance\_identifier }} is operating within normal parameters.'

summary: 'Resolved: RDS instance {{ $labels.dbinstance\_identifier }} has normal database connections'

### URL-alert.yaml >>

groups:

- name: alert.rules

rules:

- alert: EndpointDown

expr: probe\_success == 0

for: 10s

labels:

severity: "critical"

annotations:

summary: "Endpoint down"

### Goto the directory: Prometheus-grafana/reportgen and clone the below repository:

* Install ruby

sudo apt-get install ruby

* install report gen locally gem install

ruby-grafana-reporter

* create a config with the ruby generator use this command

ruby-grafana-reporter -w

* use this command to get .rb file which the container will use:

wget <https://github.com/divinity666/ruby-grafana-reporter/releases/download/v0.6.6/ruby-grafana-reporter-0.6.6.rb>

* create a sh file named startup.sh and add the below configuration:

cd /documents  
apt-get install ruby -y  
gem install ruby-grafana-reporter  
ruby-grafana-reporterh

## Start all the containers

Goto the directory where you have your docker-compose.yaml is present and run the below command:

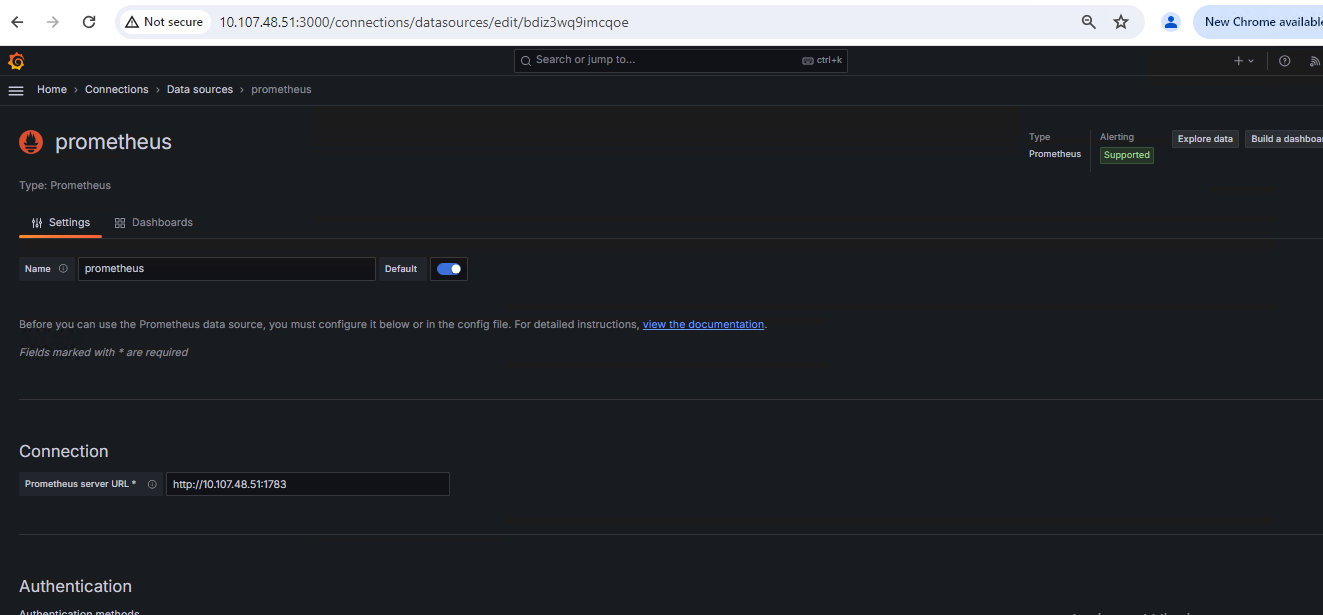
* docker compose up -d. (Try sudo if not a super-user)
* All the containers should be up and running now and you can access the services at the exposed ports.

## Prometheus:

* View the Prometheus server dashboard at http:<public\_ip>:1783 and goto targets to view all the targets.
* Confirm that all the servers are up and active (wherever you’ve installed the node exporter)

## Grafana configuration:

* Browse the Grafana Dashboard at http:<public\_ip>:3000
* Goto Datasources and add the datasource for Prometheus server. Paste the URL where your proemtheus server is running and add the datasource.



## ReportGen:

* After creating the dashboard on Grafana, go to dashboard settings and a link for where your report generator is hosted. http://<public\_ip>:8815/render?var-template=demo\_report
* Select all the checkboxes below and save the dashboard
* This will create a link to download a pdf report for the panel

