clean-monitoring

**Prometheus && Node-exporter should be on host machine**

**-----> vi /etc/docker/daemon.json**

{

"metrics-addr" : "127.0.0.1:9323",

"experimental" : true

}

**-----> vi docker-compose.yml**

version: '3.7'

services:

cadvisor:

image: gcr.io/cadvisor/cadvisor:latest

ports:

- "8080:8080"

volumes:

- "/:/rootfs"

- "/var/run:/var/run"

- "/sys:/sys"

- "/var/lib/docker/:/var/lib/docker"

- "/dev/disk/:/dev/disk"

privileged: true

devices:

- "/dev/kmsg"

depends\_on:

- redis

pushgateway:

image: prom/pushgateway

container\_name: pushgateway

ports:

- 9091:9091

#node-exporter:

# image: prom/node-exporter:latest

# container\_name: node-exporter

# restart: unless-stopped

#ports:

# - 9100:9100

grafana:

image: grafana/grafana

container\_name: grafana

ports:

- 3000:3000

environment:

- GF\_SECURITY\_ADMIN\_PASSWORD=password

depends\_on:

# - prometheus

- cadvisor

- loki

redis:

image: redis:latest

container\_name: redis

ports:

- 6379:6379

loki:

image: grafana/loki:2.9.0

container\_name: loki

ports:

- "3100:3100"

command: -config.file=/etc/loki/local-config.yaml

promtail:

image: grafana/promtail:2.9.0

container\_name: promtail

volumes:

- /var/log:/var/log

command: -config.file=/etc/promtail/config.yml

**---------> config.yml**

server:

http\_listen\_port: 9080

grpc\_listen\_port: 0

positions:

filename: /tmp/positions.yaml

clients:

- url: http://loki:3100/loki/api/v1/push

scrape\_configs:

- job\_name: system

static\_configs:

- targets:

- localhost

labels:

job: varlog

# \_\_path\_\_: /var/log/\*log

\_\_path\_\_: /var/lib/docker/containers/\*/\*log

- targets:

- localhost

labels:

job: docker

\_\_path\_\_: /var/lib/docker/containers/\*/\*log

**--------->local-config.yml**

auth\_enabled: false

server:

http\_listen\_port: 3100

grpc\_listen\_port: 9096

common:

instance\_addr: 127.0.0.1

path\_prefix: /tmp/loki

storage:

filesystem:

chunks\_directory: /tmp/loki/chunks

rules\_directory: /tmp/loki/rules

replication\_factor: 1

ring:

kvstore:

store: inmemory

query\_range:

results\_cache:

cache:

embedded\_cache:

enabled: true

max\_size\_mb: 100

schema\_config:

configs:

- from: 2020-10-24

store: boltdb-shipper

object\_store: filesystem

schema: v11

index:

prefix: index\_

period: 24h

ruler:

alertmanager\_url: http://localhost:9093

# By default, Loki will send anonymous, but uniquely-identifiable usage and configuration

# analytics to Grafana Labs. These statistics are sent to https://stats.grafana.org/

#

# Statistics help us better understand how Loki is used, and they show us performance

# levels for most users. This helps us prioritize features and documentation.

# For more information on what's sent, look at

# https://github.com/grafana/loki/blob/main/pkg/usagestats/stats.go

# Refer to the buildReport method to see what goes into a report.

#

# If you would like to disable reporting, uncomment the following lines:

#analytics:

# reporting\_enabled: false

------> vi prometheus.yml

# Sample config for Prometheus.

global:

scrape\_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.

evaluation\_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.

# scrape\_timeout is set to the global default (10s).

# Attach these labels to any time series or alerts when communicating with

# external systems (federation, remote storage, Alertmanager).

external\_labels:

monitor: 'example'

# Alertmanager configuration

alerting:

alertmanagers:

- static\_configs:

- targets: ['localhost:9093']

# Load rules once and periodically evaluate them according to the global 'evaluation\_interval'.

rule\_files:

# - "first\_rules.yml"

# - "second\_rules.yml"

# A scrape configuration containing exactly one endpoint to scrape:

# Here it's Prometheus itself.

scrape\_configs:

# The job name is added as a label `job=<job\_name>` to any timeseries scraped from this config.

- job\_name: 'prometheus'

# Override the global default and scrape targets from this job every 5 seconds.

scrape\_interval: 5s

scrape\_timeout: 5s

# metrics\_path defaults to '/metrics'

# scheme defaults to 'http'.

static\_configs:

- targets: ['localhost:9090']

- targets: ['localhost:9323']

- targets: ['localhost:9100']

- targets: ['localhost:9091']

- targets: ['localhost:8080']

- job\_name: node

# If prometheus-node-exporter is installed, grab stats about the local

# machine by default.

static\_configs:

- targets: ['localhost:9100']