

Grafana, Prometheus and Node Exporter

Erick Montes Bedolla

Abstract ▾

Create a docker-compose file with Prometheus, Grafana and Node-exporter

docker-compose.yml ▾

```
version: '3.7'

volumes:
  grafana-data:
  prometheus-data:

services:
  node-exporter:
    image: quay.io/prometheus/node-exporter:latest
    container_name: node-exporter
    restart: unless-stopped
    volumes:
      - /proc:/host/proc:ro
      - /sys:/host/sys:ro
    command:
      - '--path.procfs=/host/proc'
      - '--path.sysfs=/host/sys'
      - --collector.filesystem.ignored-mount-points
      -
    "^(sys|proc|dev|host|etc|rootfs/var/lib/docker/containers|rootfs/var/lib/docker/overlay2|rootfs/run/docker/netns|rootfs/var/lib/docker/aufs)($$|/)"
    ports:
      - 9100:9100
    expose:
      - 9100

  prometheus:
    image: prom/prometheus:latest
    container_name: prometheus
    restart: unless-stopped
    volumes:
      - ./prometheus.yml:/etc/prometheus/prometheus.yml
      - prometheus-data:/prometheus
    command:
      - '--config.file=/etc/prometheus/prometheus.yml'
      - '--storage.tsdb.path=/prometheus'
      - '--storage.tsdb.retention.time=1y'
      - '--web.enable-lifecycle'

    ports:
```

```

        - 9090:9090
    expose:
        - 9090
grafana:
    image: grafana/grafana:8.0.6
    container_name: grafana
    restart: unless-stopped
    ports:
        - 3000:3000
    volumes:
        - grafana-data:/var/lib/grafana
cadvisor:
    image: gcr.io/cadvisor/cadvisor:latest
    container_name: cadvisor
    restart: unless-stopped
    expose:
        - 8080
    volumes:
        - /:/rootfs:ro
        - /var/run:/var/run:rw
        - /sys:/sys:ro
        - /var/lib/docker:/var/lib/docker:ro
app_example:
    image: quay.io/brancz/prometheus-example-app:v0.3.0
    container_name: app_example
    restart: unless-stopped
    ports:
        - 80:8080

```

Running docker-compose ▾

In order to run the docker-compose file, write the command

```
docker-compose up -d
```

```

erick ➤ Grafana-prometheus-node-exporter ➤ → (main) ➤ ❤️ 19:10 ➤ docker-compose up -d
[+] Running 8/8
 - Network grafana-prometheus-node-exporter_default      Created
 - Volume "grafana-prometheus-node-exporter_prometheus-data" Created
 - Volume "grafana-prometheus-node-exporter_grafana-data" Created
 - Container cadvisor                                    Started
 - Container node-exporter                               Started
 - Container prometheus                                  Started
 - Container grafana                                     Started
 - Container app_example                                 Started
erick ➤ Grafana-prometheus-node-exporter ➤ → (main) ➤ ❤️ 19:10 ➤

```

We can verify the containers where created running the line

```
docker-compose ps
```

erick	Grafana-prometheus-node-exporter	→ (main)	♥ 19:10	docker-compose ps	
NAME	COMMAND	SERVICE	STATUS	PORTS	
app_example	"/bin/prometheus-exa..."	app_example	running	0.0.0.0:80->8080/tcp	
cadvisor	"/usr/bin/cadvisor -..."	cadvisor	running (healthy)	8080/tcp	
grafana	"/run.sh"	grafana	running	0.0.0.0:3000->3000/tcp	
node-exporter	"/bin/node_exporter ..."	node-exporter	running	0.0.0.0:9100->9100/tcp	
prometheus	"/bin/prometheus --c..."	prometheus	running	0.0.0.0:9090->9090/tcp	
erick	Grafana-prometheus-node-exporter	→ (main)	♥ 19:16		

We can check the ports that we specify in the docker-compose file for each container

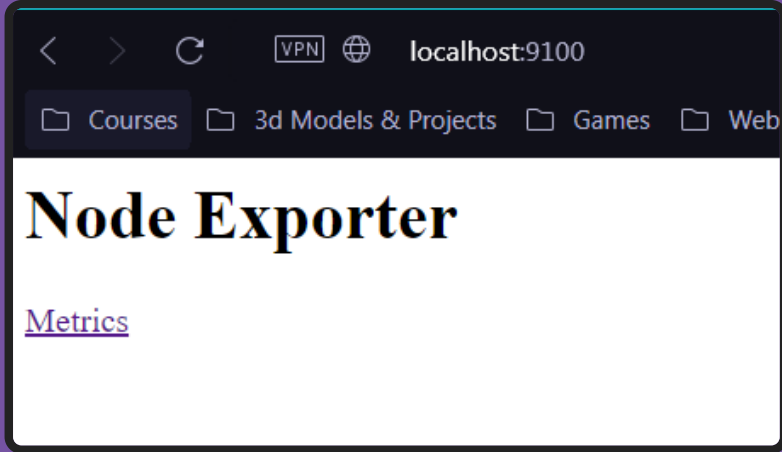
localhost:9090 for Prometheus

The screenshot shows the Prometheus web interface at localhost:9090/graph. The interface includes a search bar, navigation tabs, and a main area with the message 'No data queried yet' and an 'Add Panel' button.

localhost:3000 for Grafana

The screenshot shows the Grafana web interface at localhost:3000/explore. The interface includes a search bar, navigation tabs, and a main area with the message 'Enter a PromQL query (run with Shift+Enter)' and a 'Run query' button.

localhost:9100 for Node Exporter



```
< > ↻ [VPN] 🌐 localhost:9100/metrics
📁 Courses 📁 3d Models & Projects 📁 Games 📁 Web Design 📁 To Read 📁 To Buy 📧 Gmail-erickmontes...

# HELP go_gc_duration_seconds A summary of the pause duration of garbage collection cycles.
# TYPE go_gc_duration_seconds summary
go_gc_duration_seconds{quantile="0"} 5.0405e-05
go_gc_duration_seconds{quantile="0.25"} 5.0405e-05
go_gc_duration_seconds{quantile="0.5"} 0.000149621
go_gc_duration_seconds{quantile="0.75"} 0.000964489
go_gc_duration_seconds{quantile="1"} 0.000964489
go_gc_duration_seconds_sum 0.001164515
go_gc_duration_seconds_count 3
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 8
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.19.3"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 863536
# HELP go_memstats_alloc_bytes_total Total number of bytes allocated, even if freed.
# TYPE go_memstats_alloc_bytes_total counter
go_memstats_alloc_bytes_total 4.583304e+06
# HELP go_memstats_buck_hash_sys_bytes Number of bytes used by the profiling bucket hash table.
# TYPE go_memstats_buck_hash_sys_bytes gauge
go_memstats_buck_hash_sys_bytes 1.447795e+06
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 36021
# HELP go_memstats_gc_sys_bytes Number of bytes used for garbage collection system metadata.
```