

# Monitoring Proxmox system the easy way

We are going to be using docker and docker compose to complete this setup to work in monitoring your systems.

Here is the compose file you will need:

```
services:
  prometheus:
    image: prom/prometheus
    container_name: prometheus
    command:
      - '--config.file=/etc/prometheus/prometheus.yml'
    ports:
      - 9090:9090
    restart: unless-stopped
    extra_hosts:
      - "dockersrv002:host-gateway"
    volumes:
      - ./prometheus:/etc/prometheus
      - prom_data:/prometheus
  grafana:
    image: grafana/grafana
    container_name: grafana
    ports:
      - 3000:3000
    restart: unless-stopped
    volumes:
      - ./grafana:/etc/grafana/provisioning/datasources
      - grafana_data:/var/lib/grafana
  influxdb:
    image: influxdb:2.7.6-alpine
    container_name: influxdb2
    ports:
```

```
- 8086: 8086

extra_hosts:
  - "dockersrv002: host-gateway"

volumes:
  - influxdb2_data: /var/lib/influxdb2
  - ./influxdb2/config: /etc/influxdb2
  - ./influxdb2/scripts: /docker-entrypoint-initdb.d

volumes:
  prom_data:
  grafana_data:
  influxdb2_data:
```

# Grafana Configuration files

datasource.yml

```
apiVersion: 1

datasources:
- name: Prometheus
  type: prometheus
  url: http://prometheus:9090
  isDefault: true
  access: proxy
  editable: true
```

# Prometheus Configuration files

```
global:
  scrape_interval: 15s
  scrape_timeout: 10s
  evaluation_interval: 15s
alerting:
  alertmanagers:
  - static_configs:
    - targets: []
  scheme: http
```

```
    timeout: 10s
    api_version: v1
scrape_configs:
- job_name: prometheus
  honor_timestamps: true
  scrape_interval: 15s
  scrape_timeout: 10s
  metrics_path: /metrics
  scheme: http
  static_configs:
  - targets:
    - localhost:9090
- job_name: node_exporter
  static_configs:
  - targets:
    #- dockersrv002: 9100
    #- 192.168.10.171: 9100
    #- 192.168.30.99: 9100
```

---

Revision #5

Created 27 July 2024 21:27:32 by mbtechadmin

Updated 30 July 2024 00:34:24 by mbtechadmin