

DAY6

Prometheus is an open-source system monitoring and alerting toolkit originally built at

SoundCloud. It is now a standalone open source project . Prometheus joined the Cloud Native

Computing Foundation in 2016 as the second hosted project, after Kubernetes.

Features

1. a multi-dimensional data model with time series data identified by metric name and key/value pairs
2. PromQL, a flexible query language to leverage this dimensionality
3. no reliance on distributed storage; single server nodes are autonomous
4. time series collection happens via a pull model over HTTP
5. pushing time series is supported via an intermediary gateway
6. targets are discovered via service discovery or static configuration
7. multiple modes of graphing and dashboarding support

<https://grafana.com/dashboards/405>

<https://grafana.com/dashboards/1860>

<https://grafana.com/dashboards/5174>

<https://grafana.com/dashboards/9096>

```
rate(node_cpu_seconds_total{mode="system"}[1m])
```

```
rate(node_network_receive_bytes_total[1m])
```

node_load15

```
Setting up grafana (11.5.2) ...
info: Selecting UID from range 100 to 999...

info: Adding system user 'grafana' (UID 107) ...
info: Adding new user 'grafana' (UID 107) with group 'grafana' ...
info: Not creating home directory '/usr/share/grafana'.
### NOT starting on installation, please execute the following statements to configure grafana to start automatically using
systemd
  sudo /bin/systemctl daemon-reload
  sudo /bin/systemctl enable grafana-server
### You can start grafana-server by executing
  sudo /bin/systemctl start grafana-server
Processing triggers for man-db (2.12.0-4build2) ...
Synchronizing state of grafana-server.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service → /usr/lib/systemd/system/grafana-server.
service.
● grafana-server.service - Grafana instance
   Loaded: loaded (/usr/lib/systemd/system/grafana-server>
   Active: active (running) since Sat 2025-03-22 08:29:35>
     Docs: http://docs.grafana.org
   Main PID: 35977 ((grafana))
     Tasks: 1 (limit: 9337)
    Memory: 256.0K ()
    CGroup: /system.slice/grafana-server.service
            └─35977 "(grafana)"

Mar 22 08:29:35 DESKTOP-MJGHIPO systemd[1]: Started grafana>
ubuntu@DESKTOP-MJGHIPO:~/prometheus-2.47.1.linux-amd64$
```

```
ubundu@DESKTOP-MJGHIPO: ~$ minikube status
minikube
type: Control Plane
host: Running
kubelet: Running
apiserver: Running
kubeconfig: Configured

ubundu@DESKTOP-MJGHIPO: ~$ sudo useradd \
--system \
--no-create-home \
--shell /bin/false prometheus
[sudo] password for ubundu:
useradd: user 'prometheus' already exists
ubundu@DESKTOP-MJGHIPO: ~$ wget https://github.com/prometheus/prometheus/releases/download/v2.47.1/prometheus-2.47.1.linux-amd64.tar.gz
tar -xvf prometheus-2.47.1.linux-amd64.tar.gz
sudo mkdir -p /data /etc/prometheus
cd prometheus-2.47.1.linux-amd64/
sudo mv prometheus promtool /usr/local/bin/
sudo mv consoles/ console_libraries/ /etc/prometheus/
sudo mv prometheus.yml /etc/prometheus/prometheus.yml
sudo chown -R prometheus:prometheus /etc/prometheus/ /data/
--2025-03-22 08:20:55-- https://github.com/prometheus/prometheus/releases/download/v2.47.1/prometheus-2.47.1.linux-amd64.tar.gz
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e6d6...
82fd743?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F...
X-Amz-Date=20250322T082056Z&X-Amz-Expires=300&X-Amz-Signature=8b1e12d49cf5eced2941f...
```

localhost:9090/targets?search=

Prometheus Alerts Graph Status Help

Targets

All scrape pools All Unhealthy Collapse All Filter by endpoint or labels

Unknown Unhealthy Healthy

jenkins (0/1 up) show logs

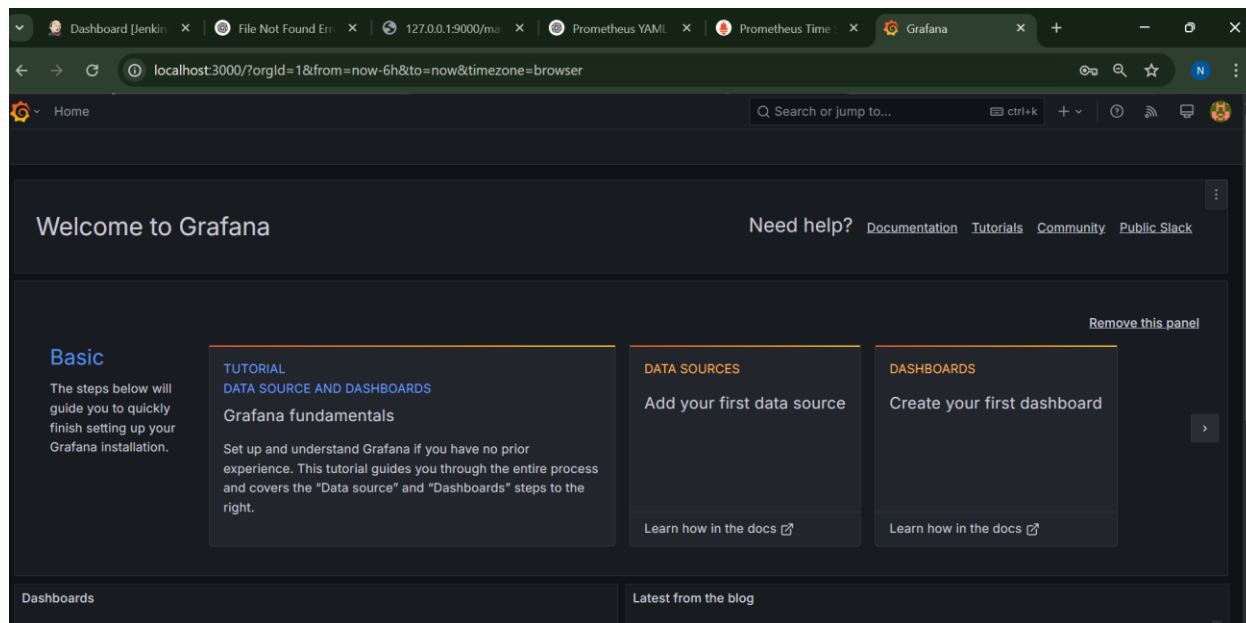
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://<jenkins-ip>:8080/prometheus	DOWN	instance="<jenkins-ip>:8080" job="jenkins"	394.000ms ago	0.220ms	Get "http://<jenkins-ip>:8080/prometheus": dial tcp: lookup <jenkins-ip>: no such host

node export (1/1 up) show logs

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9100/metrics	UP	instance="localhost:9100" job="node_export"	11.233s ago	19.841ms	

prometheus (1/1 up) show logs

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9090/metrics	UP	instance="localhost:9090" job="prometheus"	10.281s ago	8.530ms	




```
# 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"} 5.9879e-05
go_gc_duration_seconds{quantile="0.25"} 0.000146969
go_gc_duration_seconds{quantile="0.5"} 0.000187749
go_gc_duration_seconds{quantile="0.75"} 0.00035961
go_gc_duration_seconds{quantile="1"} 0.00135097
go_gc_duration_seconds_sum 0.009039947
go_gc_duration_seconds_count 31
# HELP go_goroutines Number of goroutines that currently exist.
# TYPE go_goroutines gauge
go_goroutines 36
# HELP go_info Information about the Go environment.
# TYPE go_info gauge
go_info{version="go1.21.1"} 1
# HELP go_memstats_alloc_bytes Number of bytes allocated and still in use.
# TYPE go_memstats_alloc_bytes gauge
go_memstats_alloc_bytes 2.5360568e+07
# 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 1.84000352e+08
# 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 1.492327e+06
# HELP go_memstats_frees_total Total number of frees.
# TYPE go_memstats_frees_total counter
go_memstats_frees_total 1.302908e+06
# HELP go_memstats_gc_sys_bytes Number of bytes used for garbage collection system metadata.
# TYPE go_memstats_gc_sys_bytes gauge
go_memstats_gc_sys_bytes 4.830976e+06
# HELP go_memstats_heap_alloc_bytes Number of heap bytes allocated and still in use.
# TYPE go_memstats_heap_alloc_bytes gauge
go_memstats_heap_alloc_bytes 2.5360568e+07
# HELP go_memstats_heap_idle_bytes Number of heap bytes waiting to be used.
# TYPE go_memstats_heap_idle_bytes gauge
go_memstats_heap_idle_bytes 1.0903552e+07
# HELP go_memstats_heap_inuse_bytes Number of heap bytes that are in use.
# TYPE go_memstats_heap_inuse_bytes gauge
go_memstats_heap_inuse_bytes 2.9696e+07
# HELP go_memstats_heap_objects Number of allocated objects.
# TYPE go_memstats_heap_objects gauge
go_memstats_heap_objects 116081
# HELP go_memstats_heap_released_bytes Number of heap bytes released to OS.
# TYPE go_memstats_heap_released_bytes gauge
go_memstats_heap_released_bytes 4.292608e+06
# HELP go_memstats_heap_sys_bytes Number of heap bytes obtained from system.
# TYPE go_memstats_heap_sys_bytes gauge
go_memstats_heap_sys_bytes 4.0599552e+07
```



Search



localhost:3000/connections/datasources/edit/aeglebdjletxcc

 Home > Connections > Data sources > prometheus

Cache level ⓘ Low

Incremental querying (beta) ⓘ ☐

Disable recording rules (beta) ⓘ ☐

Other

Custom query parameters ⓘ Example: max_source_resolution=5m&timeout

HTTP method ⓘ POST

Use series endpoint ⓘ ☐

Exemplars

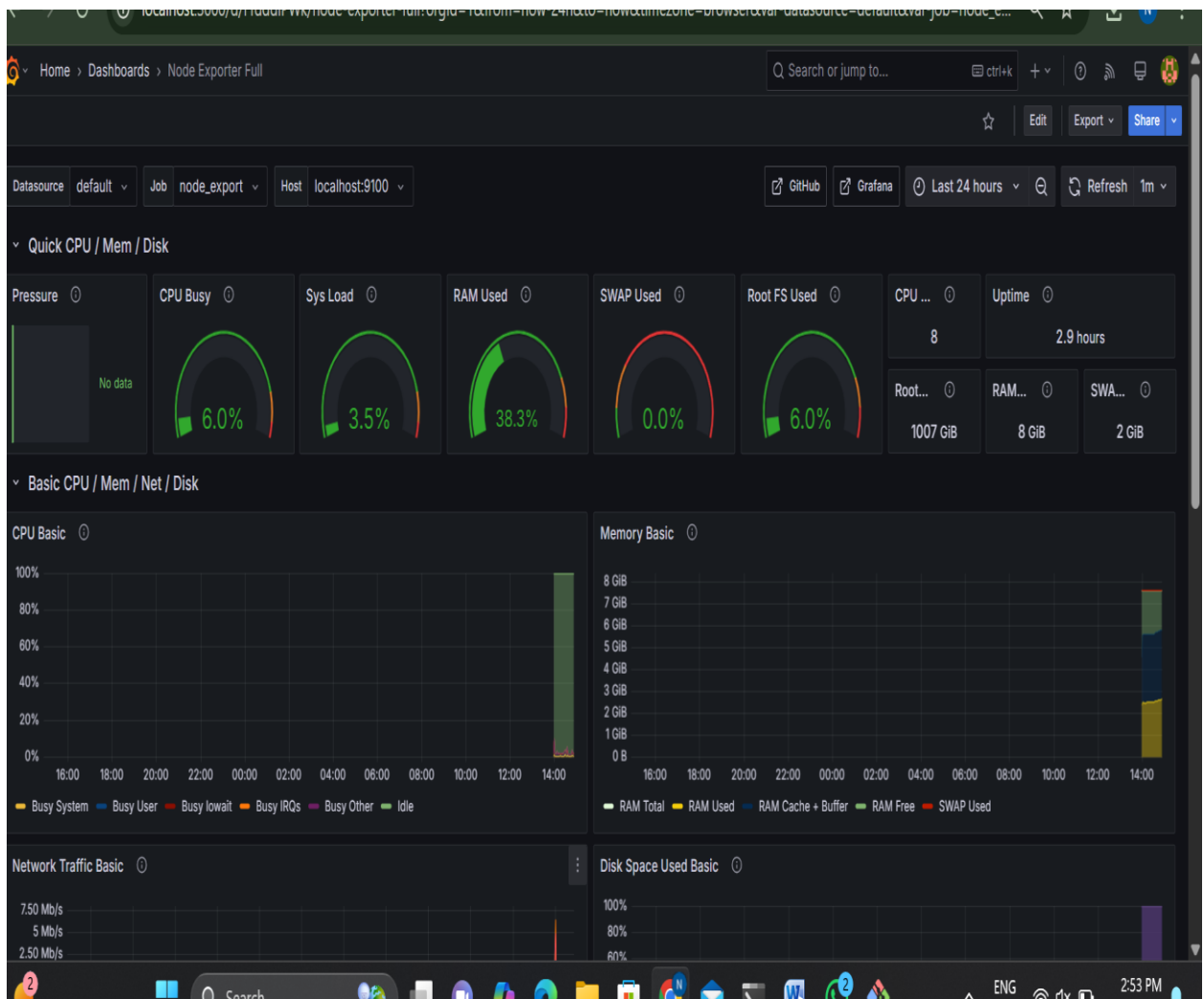
+ Add

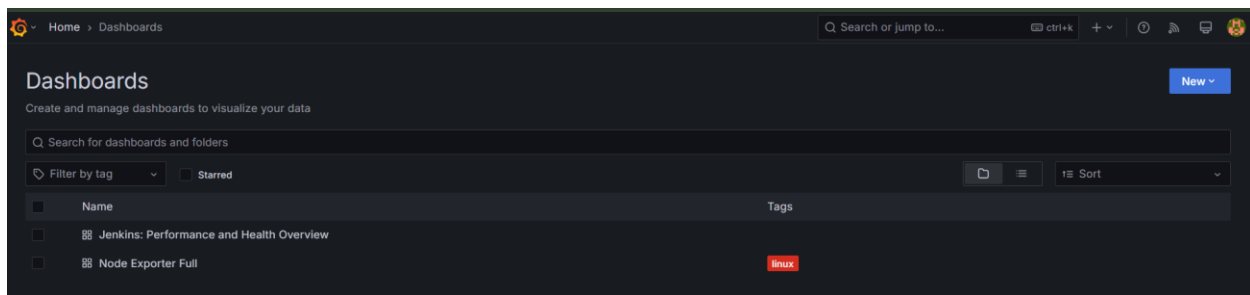
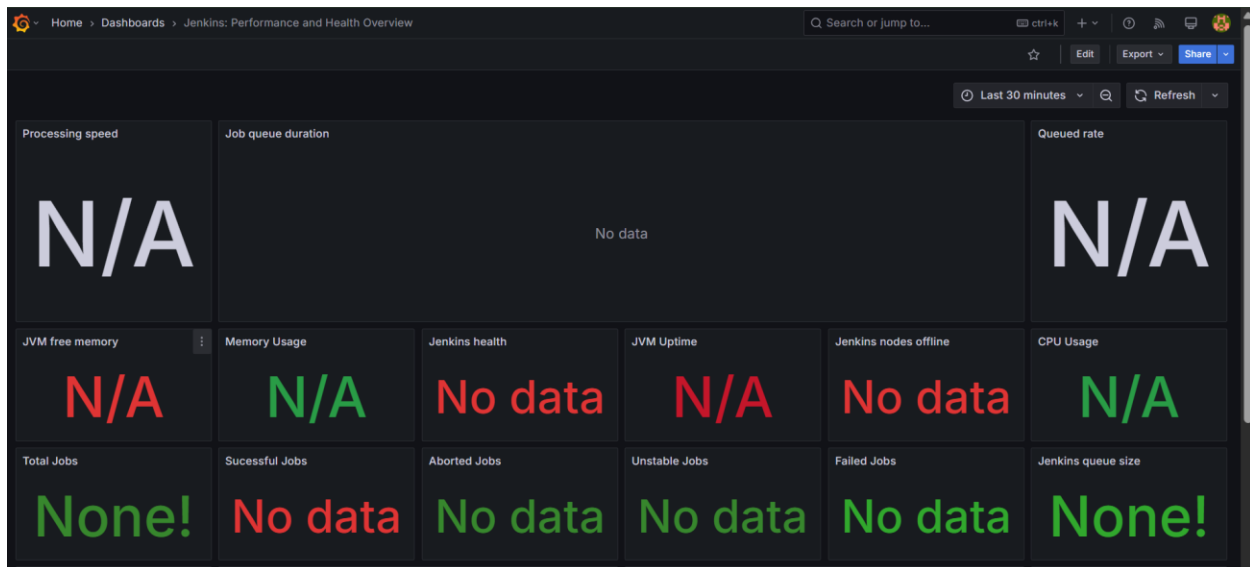
✓ Successfully queried the Prometheus API.

Next, you can start to visualize data by [building a dashboard](#), or by querying data in the [Explore view](#).

Delete

Save & test







localhost:3000/d/KvM1xSCZz/1-node-exporter-1-0-1?orgId=1&

Home > Dashboards > 1 Node Exporter 1.0.1

Datasource prometheus ▾

Interval 5m ▾

Env ▾

Host All × × ▾

> Overview (5 panels)

> Kernel (4 panels)

> Entropy (1 panel)

> Load (4 panels)

> CPU (2 panels)

> Memory (2 panels)

> Disk (9 panels)

> Filesystem (3 panels)

> Descriptors (2 panels)

> Network (10 panels)

> NTP (3 panels)

> Processes (3 panels)

