

```
+++ title = "With Grafana and Prometheus" description = "Guide for getting
started with Grafana and Prometheus" keywords = ["grafana", "intro", "guide",
"started"] aliases = ["/docs/grafana/latest/guides/gettingstarted", "/docs/grafana/latest/guides/getting_started
weight = 300 +++
```

Getting started with Grafana and Prometheus

Prometheus is an open source monitoring system for which Grafana provides out-of-the-box support. This topic walks you through the steps to create a series of dashboards in Grafana to display system metrics for a server monitored by Prometheus.

Note: You can configure a Grafana Cloud instance to display system metrics without having to host Grafana yourself. A free forever plan provides 10,000 active series for metrics.

```
{{< docs/shared "getting-started/first-step.md" >}}
```

Step 2. Download Prometheus and node_exporter

Prometheus, like Grafana, can be installed on many different operating systems. Refer to the Prometheus download page, which lists all stable versions of Prometheus components. Download the following components:

- Prometheus
- node_exporter

Step 3. Install Prometheus node_exporter

Prometheus node_exporter is a widely used tool that exposes system metrics. Install node_exporter on all hosts you want to monitor. For instructions on how to install node_exporter, refer to the Installing and running the node_exporter section in Prometheus documentation.

Note: The instructions in the referenced topic are intended for Linux users. You may have to alter the instructions slightly depending on your operating system. For example, if you are on Windows, use the windows_exporter instead.

Step 4. Install and configure Prometheus

1. Install Prometheus following instructions in the Installation topic in the Prometheus documentation.
2. Configure Prometheus to monitor the hosts where you installed node_exporter. In order to do this, modify Prometheus's configuration file. By default, Prometheus looks for the file `prometheus.yml` in the current working directory. This behavior can be changed via the `--config.file`

command line flag. For example, some Prometheus installers use it to set the configuration file to `/etc/prometheus/prometheus.yml`. Here is an example of the code you will need to add.

```
# A scrape configuration containing exactly one endpoint to scrape from node_exporter
scrape_configs:
  # The job name is added as a label `job=<job_name>` to any timeseries scraped from
  - job_name: 'node'

  # metrics_path defaults to '/metrics'
  # scheme defaults to 'http'.

static_configs:
  - targets: ['<hostname>:9100']
```

3. Start the Prometheus service:

```
./prometheus --config.file=./prometheus.yml
```

Step 5. Check Prometheus metrics in Grafana Explore view

In your Grafana instance, go to the [Explore]({{< relref “../explore/_index.md” >}}) view and build queries to experiment with the metrics you want to monitor. Here you can also debug issues related to collecting metrics from Prometheus. Pay special attention to the [Prometheus-specific features]({{< relref “../explore/_index.md#prometheus-specific-features” >}}) to avail custom querying experience for Prometheus.

Step 6. Start building dashboards

Now that you have a curated list of queries, create [dashboards]({{< relref “../dashboards/_index.md” >}}) to render system metrics monitored by Prometheus. When you install Prometheus and `node_exporter` or `windows_exporter`, you will find recommended dashboards for use.

The following image shows a dashboard with three panels showing some system metrics.

To learn more:

- Grafana documentation: [Prometheus data source]({{< relref “../data-sources/prometheus.md” >}})
- Prometheus documentation: What is Prometheus?