**Introduction**

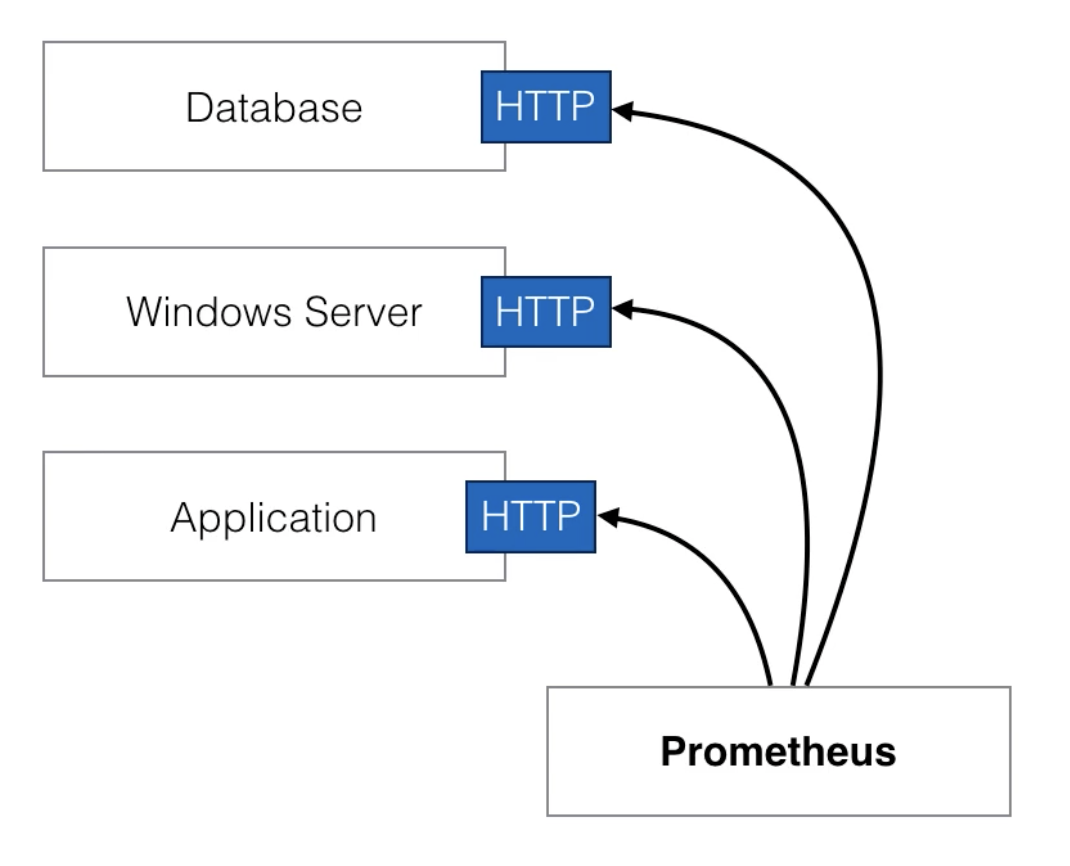
* prometheus is an open-source monitoring solution
* started at around 2012-2013 and was made public in early 2015
* prometheus provides metrics & alerting
* it's inspired by Google's Borgmon, which uses time-series data as source, and alert based on this data
* prometheus considers as best fit for cloud native infrastructure
* in prometheus we talk about Dimensional data
  + time series are identified by metric name and set of key/value pairs
  + example:

| **Metric name** | **Label** | **Sample** |
| --- | --- | --- |
| Temperature | location | 90 |

* prometheus also has flexible query language
* visualization can be shown using a built-in browser or with integrations like Grafana
* it stores metrics in memory and localdisk in an own custom, efficient format
* prometheus is completely written in Go programming language
* prometheus also allows many client libraries and integrations which makes it more flexible

**How does promethus work?**

* prometheus collects metrics from targets by scrapping using HTTP protocol

[](https://github.com/vijayprabhu04/wic-devops-mar-22/blob/main/9-monitoring/images/image_1.png)

* scrapping endpoints is much more efficient then other mechanisams
* For instance, one single prometheus server can ingest upto one million samples per seconds

**Prometheus installation**

* Prometheus can be installed in any modern linux distributions
* Prometheus can be installed on virtual machine or even it can be configured on docker containers

**How I am going to install prometheus?**

* I will be configuring prometheus on ubuntu vm & I will be using script to install prometheus
* you can also download full distribution from offical website <https://github.com/prometheus/prometheus> if incase you don't want to install through script

**Follow the below steps to install and setup:**

**🡪**Take ubuntu 18.04 and medium configuration server

🡪create script as scriptname.sh

🡪In script mention all the steps to install Prometheus

🡪Modify the permission

[]chmod 777 scriptname

🡪Run the script

[]./scriptname

🡪Go to browser and check

Ip:9090

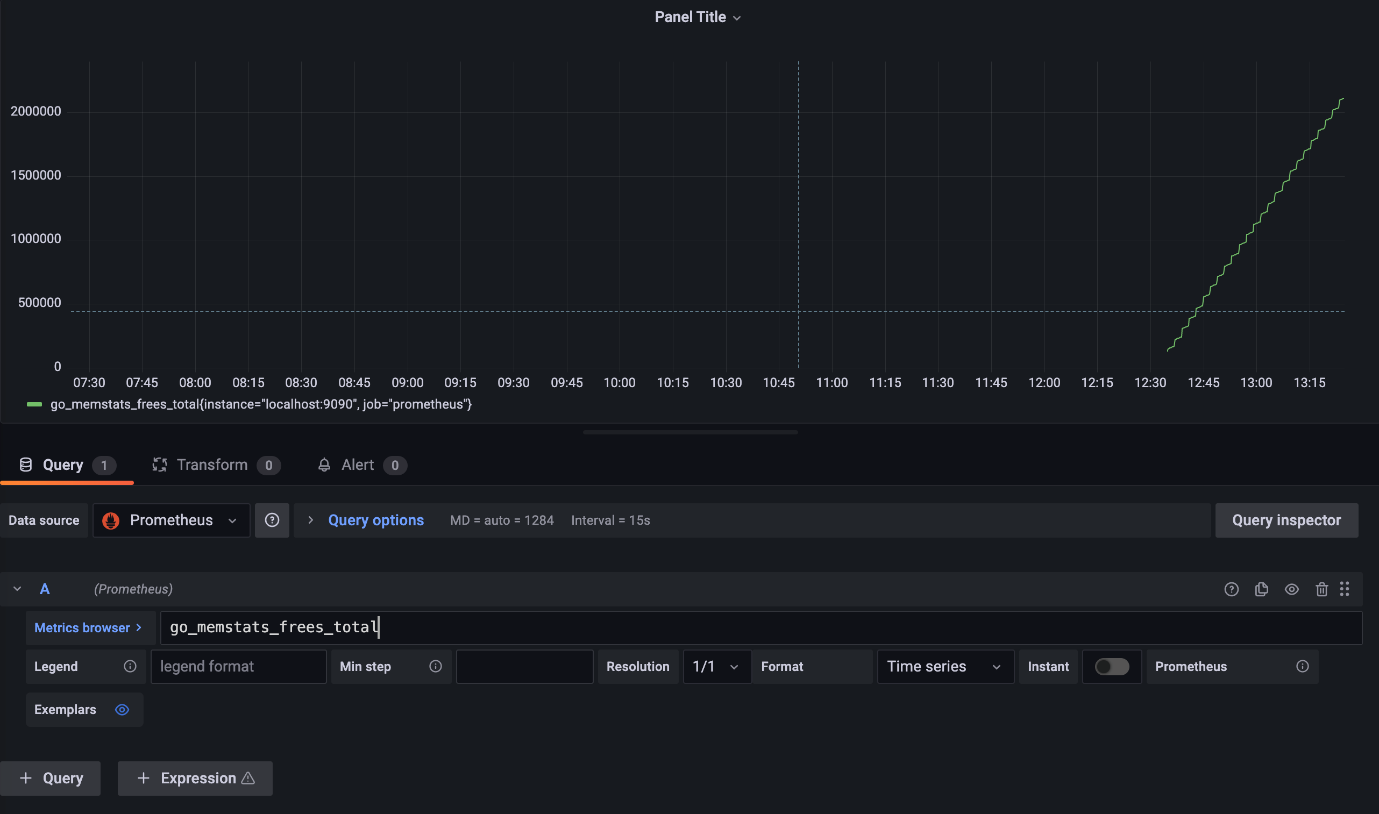
.

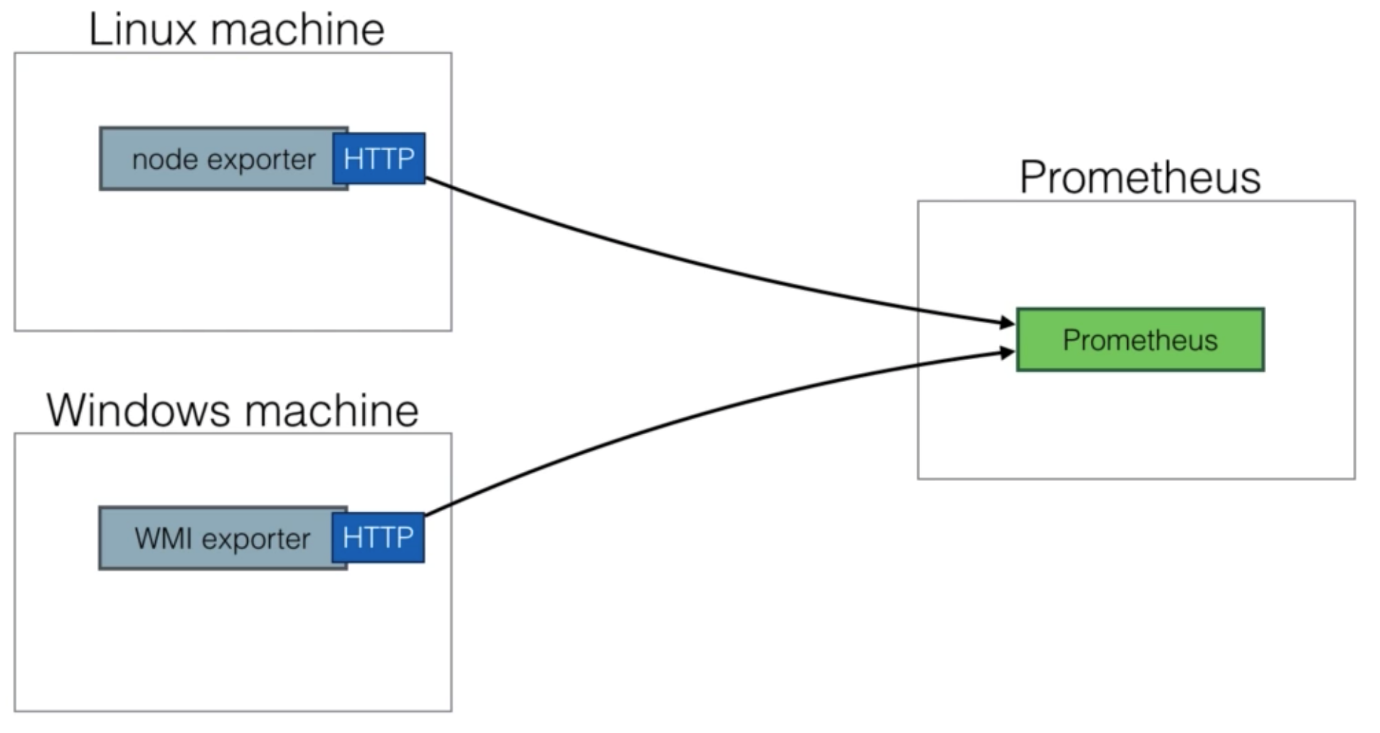
**How to check if prometheus is installed?**

* ps aux | grep promotheus
  + check if you are able to see the running process for prometheus
* check if you are able to access inbuid prometheus dashboard

http://<ipaddress>:9090

* you can select any metrics and view it in form of graph





# Grafana

* As we saw prometheus has pretty basic dashboard everyone tend to use Grafana along with prometheus as mainstream dashboard
* Grafana enables nicer dashboard and better visualizations

# Grafana installation

* Install grafana using script in scripts folder

# Validation post install

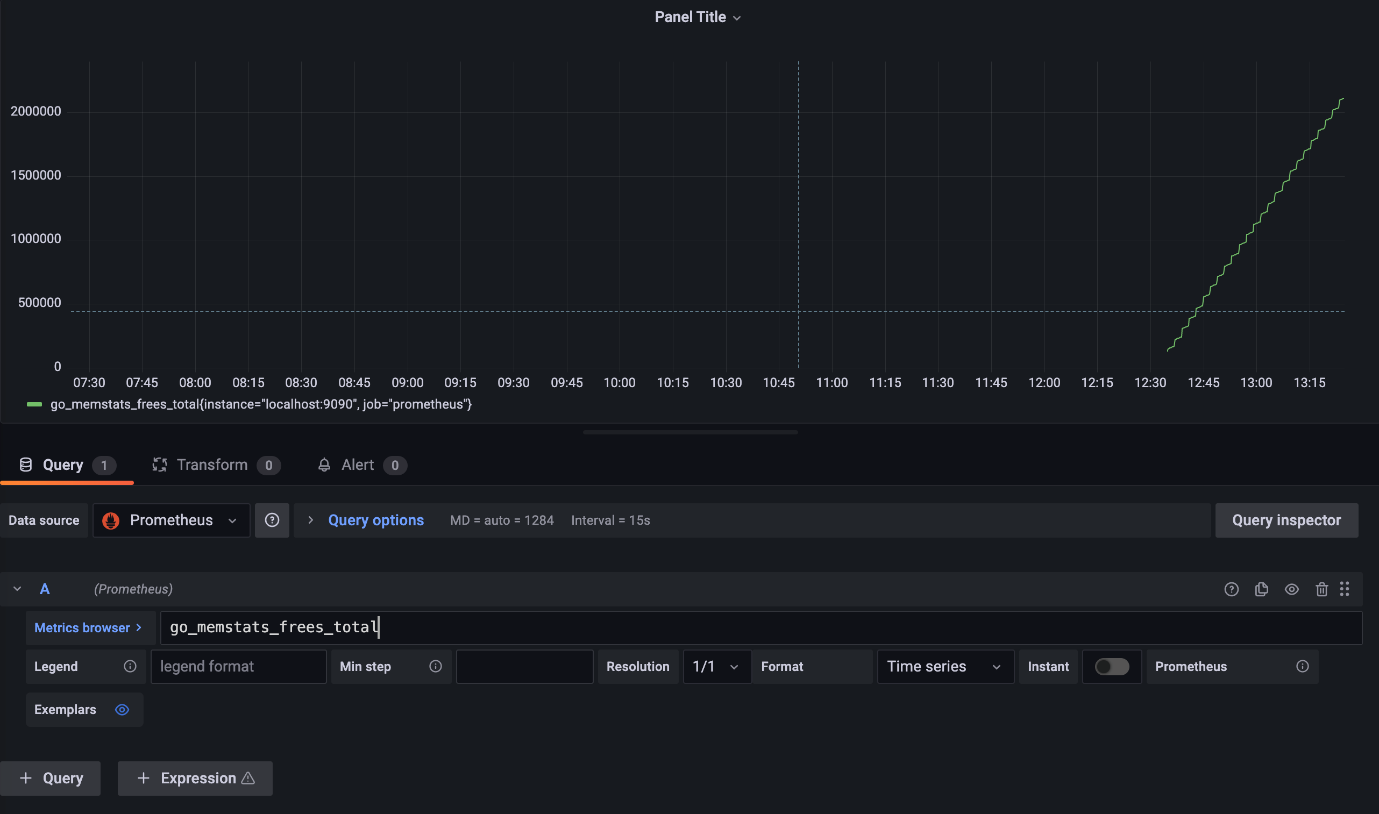
* check if grafana is installed
* ps aux | grep grafana
* http://:3000
* default username/password : admin/admin

# Add data source

* select add datasource from grafana dashboard
* select prometheus as datasource
* add url which is pointing to local prometheus datasource
* [http://localhost:9090](http://localhost:9090/)

# Add dashboard

* click on add dash board
* select prometheus from query datasource
* copy any metrics from prometheus dashboard and paste here to get graphs

[](https://github.com/vijayprabhu04/wic-devops-mar-22/blob/main/9-monitoring/images/image_2.png)

🡪Setup Grafana in same machine where Prometheus got installed

🡪Run the Grafana configuration file

🡪Hit from the browser with 3000 ports

🡪Create dashboard and import Prometheus logs

**Setting up ec2 server and checking the logs from Grafana dashboard:**

🡪Create ec2 instance and do the following changes

🡪Run the node exporter.sh file in this machine

🡪At end it is giving few lines. Copy those line and paste in Grafana server

🡪Path to add lines in Grafana server

**Prometheus config file**

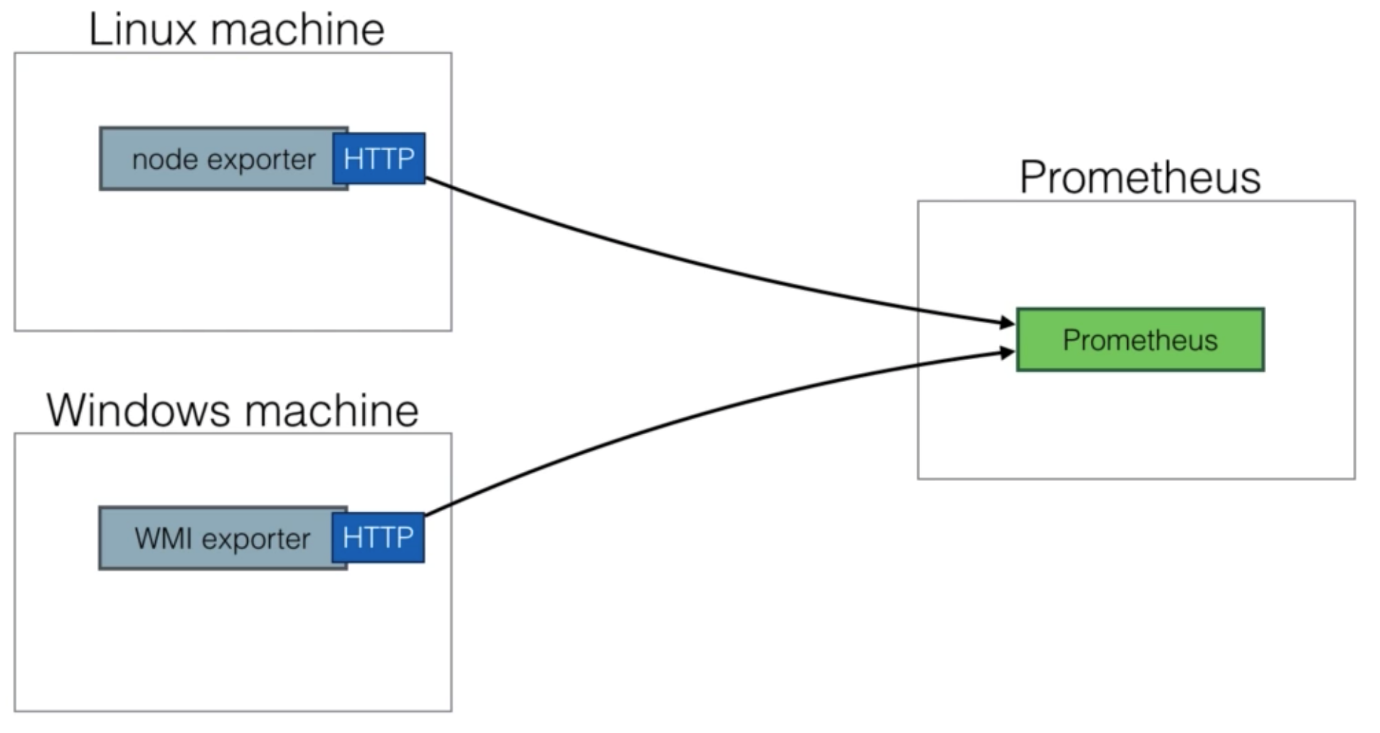
* You can find prometheus config file under /etc/prometheus/prometheus.yml
* If any changes to be done on prometheus setting to be done from this file
* There are the line to be notes
* scrape\_configs:
* # The job name is added as a label `job=<job\_name>` to any timeseries scraped from this config.
* - job\_name: 'prometheus'
* # metrics\_path defaults to '/metrics'
* # scheme defaults to 'http'.
* static\_configs:
* - targets: ['localhost:9090']
* This lines are actually shipping localhost metrics
* You can also find this in targets under status in prometheus dashboard

# Node exporter

* To monitor nodes, you need to install the node exporter
* The node exporter will expose machine metrics of linux machines

For example : cpu usage, memory usage

* Node exporter can be used to monitor machines & later on, you can create alerts based on ingested metrics
* For windows, there's a WMI exporter to do the same task

[](https://github.com/vijayprabhu04/wic-devops-mar-22/blob/main/9-monitoring/images/image_3.png)

# Node exporter install

* use node exporter script node-exporter.sh to install node exporter in target machines
* once node exporter is installed in end point just copy job and add under your prometheus.yml file

# Reload prometheus

* kill -HUP <process-number>
* once done you should be able to see new exporter under targets
* you can either view it in prometheus dashboard or you can create new dashboard in grafana