**Install Prometheus and Grafana on Ubuntu**

**1.Creating Prometheus System Users and Directory**

1 Prometheus can be started and stopped from the command line, it is more convenient to run it as a service using the systemctl utility. This allows it to run in the background.

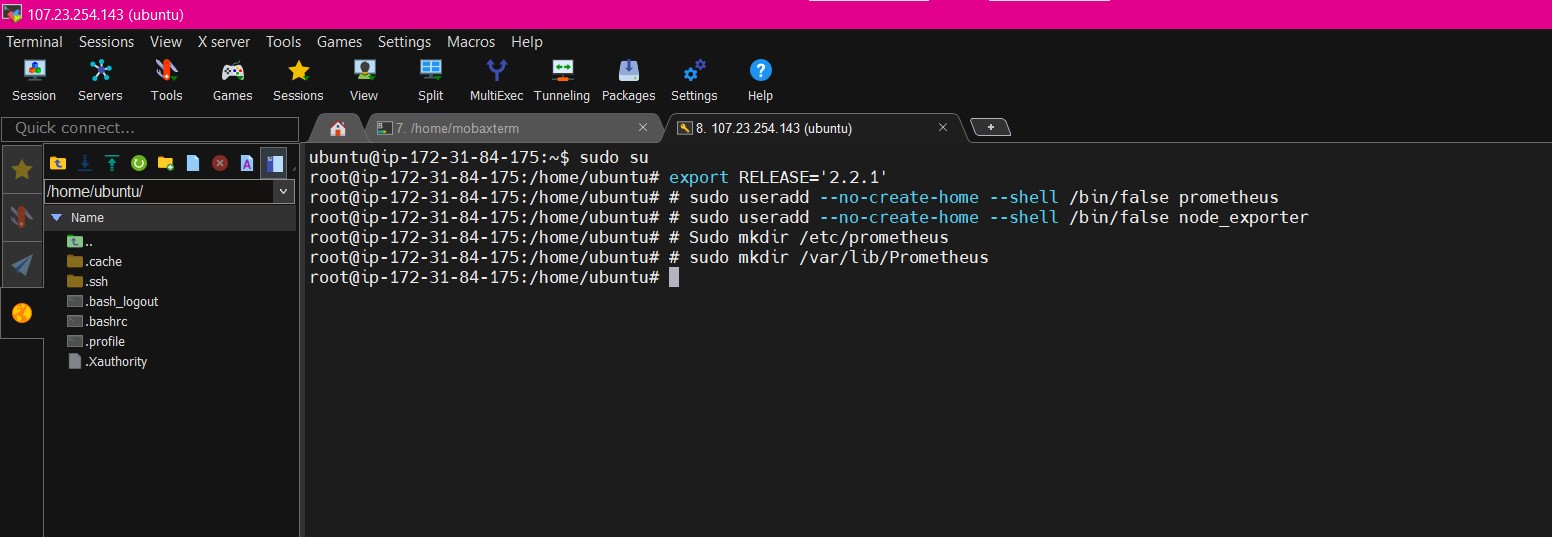
2 .We will have to create a Prometheus user named Prometheus and a Prometheus directory named as Prometheus.

# sudo useradd --no-create-home --shell /bin/false prometheus

# sudo useradd --no-create-home --shell /bin/false node\_exporter

# Sudo mkdir /etc/prometheus

# sudo mkdir /var/lib/Prometheus



**2.Update Prometheus user**

As user groups and directories are created successfully which store the Prometheus data and files. Now we will have to update the group and user ownership on the newly created directories. By using the below command we update the ownership.

# sudo chown prometheus:prometheus /etc/prometheus

# sudo chown prometheus:prometheus /var/lib/Prometheus

3.Download Prometheus Binary File

Now we will download the latest version of Prometheus. And past in /opt navigate to /opt directory

# cd /opt/

Download the Prometheus setup using wget

# wget https://github.com/prometheus/prometheus/releases/download/v2.26.0/prometheus-2.26.0.linux-amd64.tar.gz

Now we have successfully downloaded the Prometheus file and now we will extract that file.

4.Install Prometheus

Now we will extract the Prometheus setup file using the following commands.

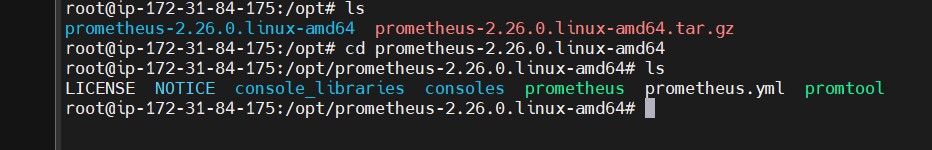
# tar -xvf prometheus-2.26.0.linux-amd64.tar.gz

navigate to prometheus extracted folder

# cd prometheus-2.26.0.linux-amd64

To check list of setup files

# ls



5.Copy Prometheus Binary files

Now we have two libraries in our directory i.e. Prometheus and promtool. We will have to copy that both libraries to our /usr/local/bin directory.

By following below commands we will perform the copy operation.

# sudo cp /opt/prometheus-2.26.0.linux-amd64/prometheus /usr/local/bin/

# sudo cp /opt/prometheus-2.26.0.linux-amd64/promtool /usr/local/bin/

6.Update Prometheus user ownership on Binaries

Now we will update the user and group ownership on the binaries of Prometheus.

Using following commands we will update the user and group ownership.

# sudo chown prometheus:prometheus /usr/local/bin/prometheus

# sudo chown prometheus:prometheus /usr/local/bin/promtool

7.Copy Prometheus Console Libraries

We will need to copy the console and console\_libraries directories to /etc/Prometheus/.

Use below commands to copy console and console\_libraries.

#sudo cp -r /opt/prometheus-2.26.0.linux-amd64/consoles /etc/prometheus

#sudo cp -r /opt/prometheus-2.26.0.linux-amd64/console\_libraries /etc/prometheus

#sudo cp -r /opt/prometheus-2.26.0.linux-amd64/prometheus.yml /etc/prometheus

8.Update Prometheus ownership on Directories

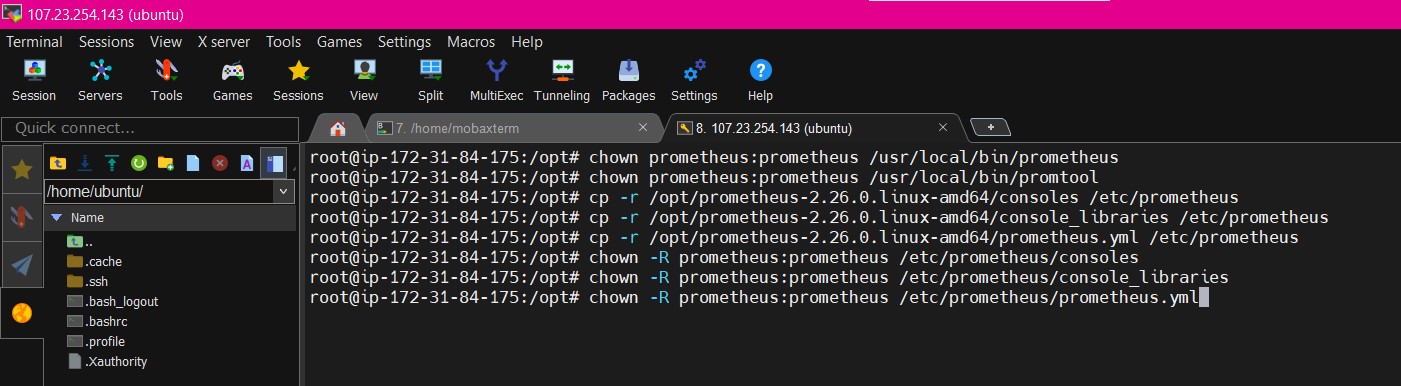
Now we will update the user and group ownership on the directories to Prometheus user using -R.

By executing this commands ownership is set on is ensured. Execute the following commands.

# sudo chown -R prometheus:prometheus /etc/prometheus/consoles

# sudo chown -R prometheus:prometheus /etc/prometheus/console\_libraries

# sudo chown -R prometheus:prometheus /etc/prometheus/prometheus.yml

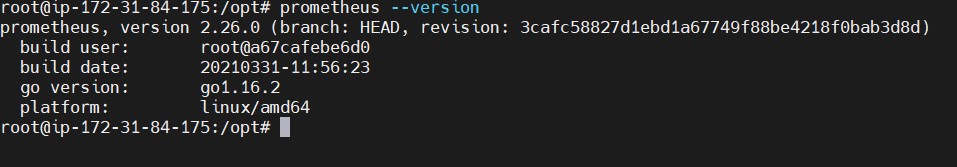


9.Check Prometheus Version

#prometheus --version

#promtool –version

Optput :



10.Prometheus configuration file

We have already copied /opt/prometheus-2.26.0.linux-amd64/prometheus.yml file /etc/prometheus directory, verify if it present and should look like below and modify it as per your requirement.

# vim /etc/prometheus/prometheus.yml

|  |
| --- |
| # my global config  global:  scrape\_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1 minute.  evaluation\_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.  # scrape\_timeout is set to the global default (10s).  # Alertmanager configuration  alerting:  alertmanagers:  - static\_configs:  - targets:  # - alertmanager:9093  # Load rules once and periodically evaluate them according to the global 'evaluation\_interval'.  rule\_files:  # - "first\_rules.yml"  # - "second\_rules.yml"  # A scrape configuration containing exactly one endpoint to scrape:  # Here it's Prometheus itself.  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'] |

#11.Creating Prometheus Systemd file

To run Prometheus as service we have to setting up prometheus, We will provide a path for both configuration file and data directory. We will start it with the Prometheus user using the following command.

# sudo -u prometheus /usr/local/bin/prometheus \

--config.file /etc/prometheus/prometheus.yml \

--storage.tsdb.path /var/lib/prometheus/ \

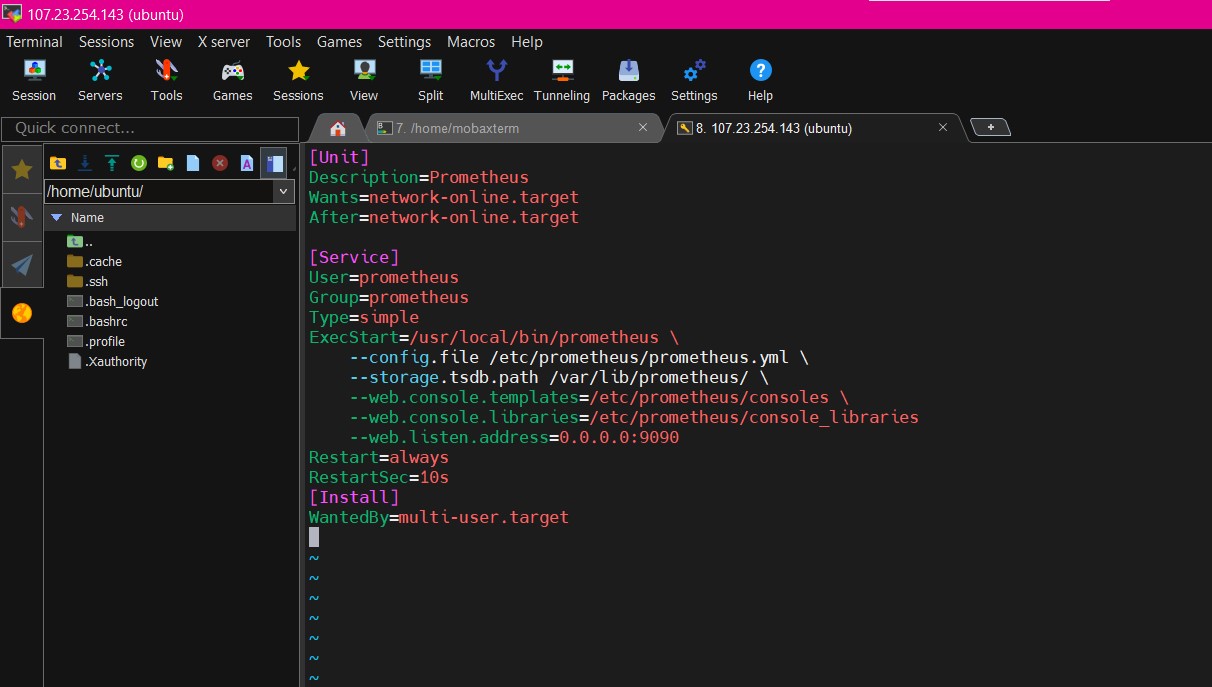
--web.console.templates=/etc/prometheus/consoles \

--web.console.libraries=/etc/prometheus/console\_libraries

Now we will create a system service file in /etc/systemd/system location

# sudo vim/etc/systemd/system/prometheus.service

|  |
| --- |
| [Unit]  Description=Prometheus  Wants=network-online.target  After=network-online.target  [Service]  User=prometheus  Group=prometheus  Type=simple  ExecStart=/usr/local/bin/prometheus \  --config.file /etc/prometheus/prometheus.yml \  --storage.tsdb.path /var/lib/prometheus/ \  --web.console.templates=/etc/prometheus/consoles \  --web.console.libraries=/etc/prometheus/console\_libraries  --web.listen.address=0.0.0.0:9090  Restart=always  RestartSec=10s  [Install]  WantedBy=multi-user.target |



To use the newly created service we will have to reload the daemon services, Use the below command to reload daemon services.

# sudo systemctl daemon-reload

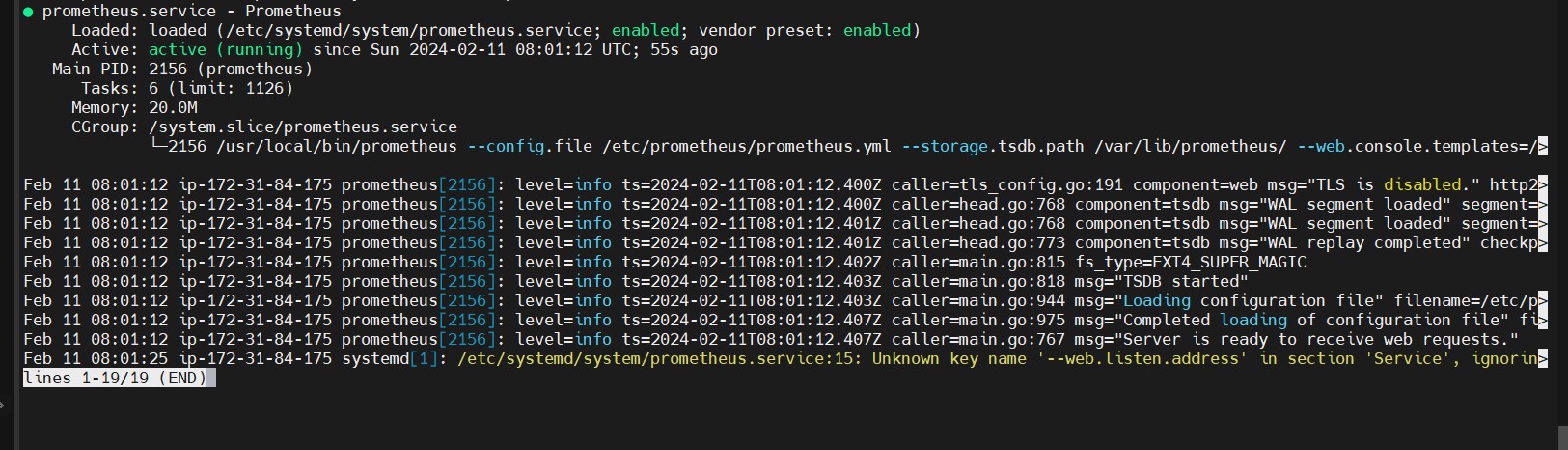
start and enable prometheus service using below commands

#sudo systemctl start prometheus

#sudo systemctl enable prometheus

We will check the Prometheus status weather it is running or not

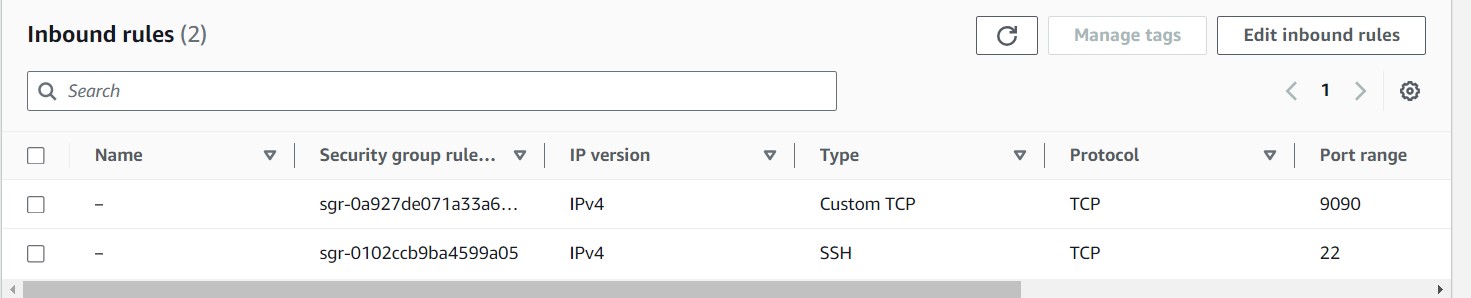
# sudo systemctl status Prometheus



12.Accessing Prometheus

Now Prometheus service is ready to run and we can access it from any web browser

NOTE- update security group rule to prot 9090:



<http://server-IP-or-Hostname:9090>.

#13.Install Grafana on Ubuntu

Prometheus is now collecting statistics from the clients listed in the scrape\_configs section of its configuration file. However, the information can only be viewed as a raw data dump. The statistics are difficult to read and not too useful.

Grafana provides an interface for viewing the statistics collected by Prometheus. Install Grafana on the same server running Prometheus and add Prometheus as a data source. Then install one or more panels for interpreting the data. To install and configure Grafana, follow these steps.

Add the Grafana GPG key in Ubuntu using wge

#wget -q -O - https://packages.grafana.com/gpg.key | sudo apt-key add –

Now add the Grafana repository in Ubuntu using APT

# echo "deb https://packages.grafana.com/oss/deb stable main" | sudo tee -a /etc/apt/sources.list.d/grafana.list

update the system packages to take effect

# sudo apt-get update

Now lets Install Grafana using below command

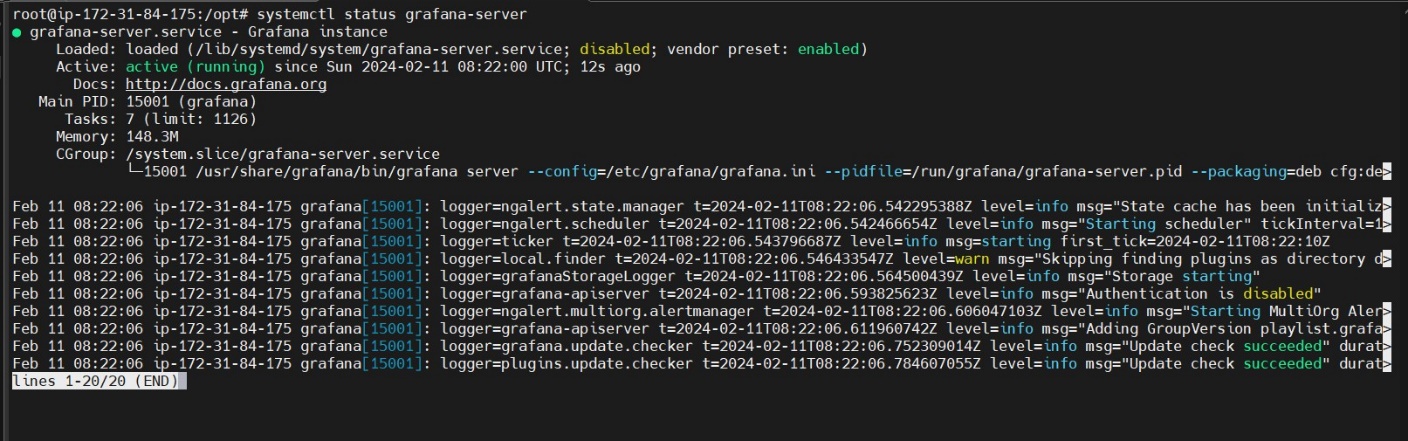
# sudo apt-get install grafana

Now start the Grafana service using below command

# sudo systemctl start grafana-server

Verify the Grafana Service Status using below command

# sudo systemctl status grafana-server

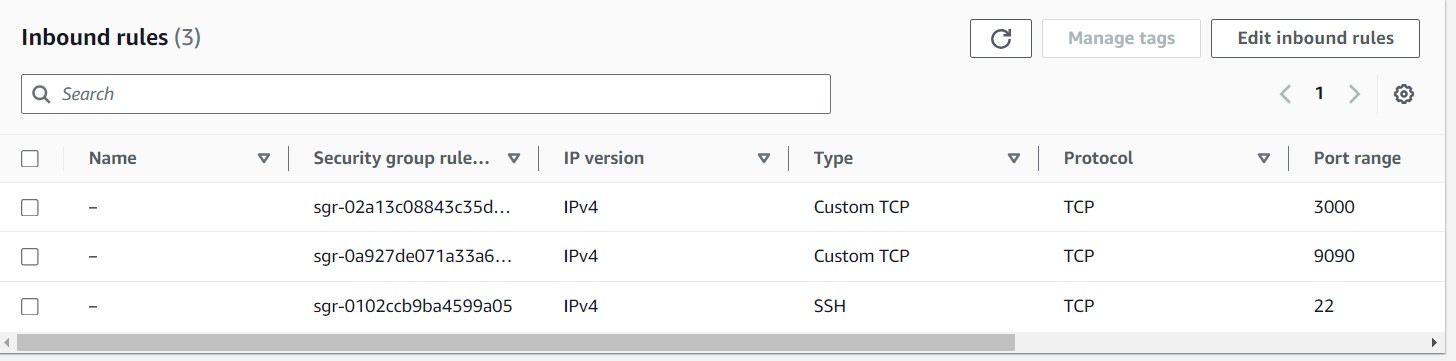


Now finally enable the Grafana service which will automatically start the Grafana on boot

#sudo systemctl enable grafana-server.service

To access Grafana Dashboard open your favorite browser, type server IP or Name followed by grafana default port 3000.

NOTE – Again add security group inbound rule to 3000

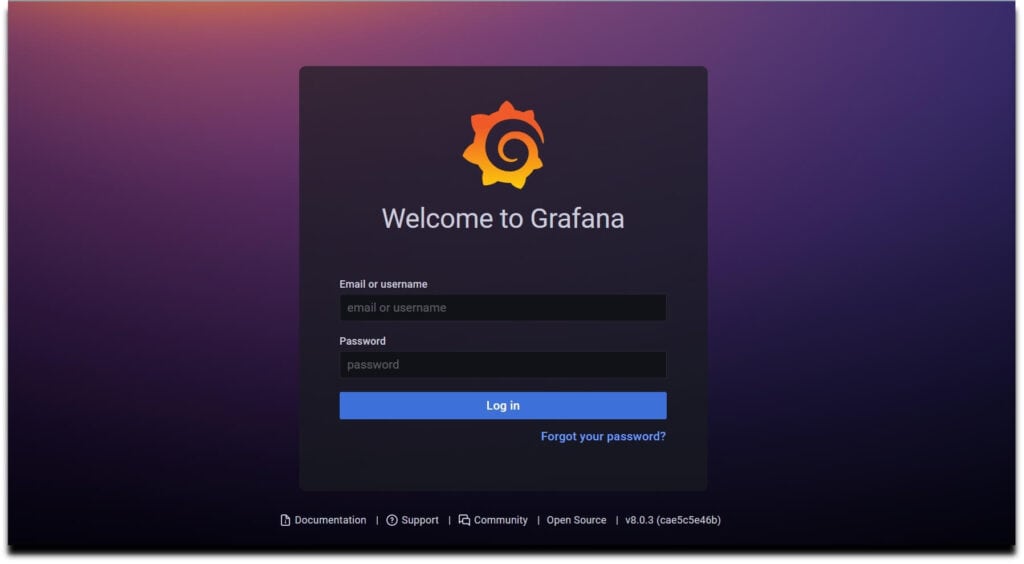


# http://your\_ip:3000

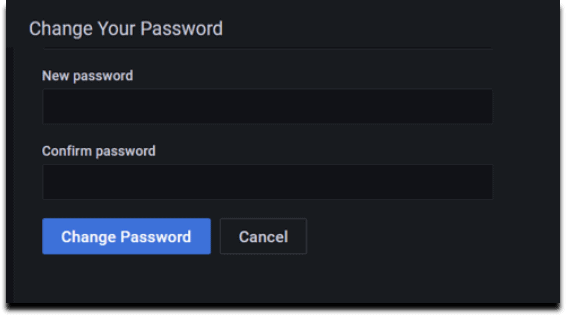
Here you can see Login page of Grafana now you will have to login with below Grafana default UserName and Password.

Username – admin

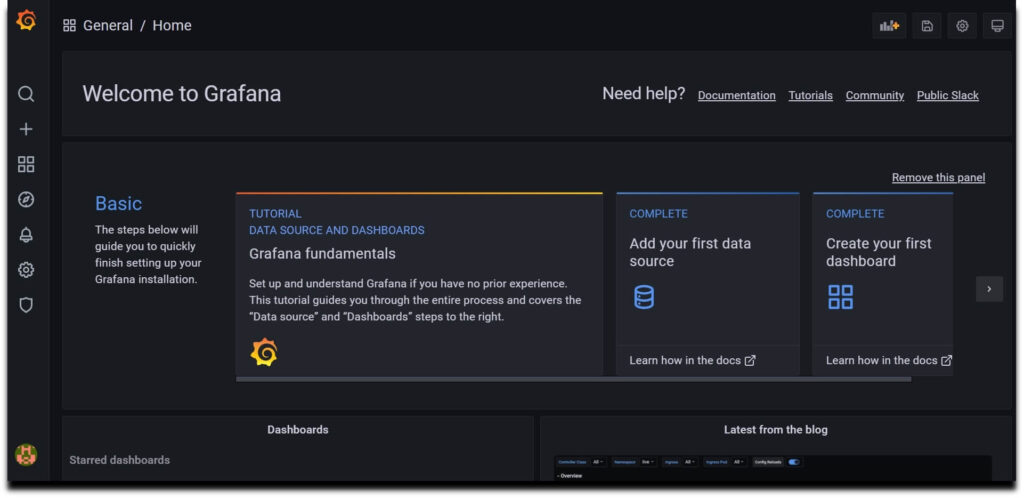
Password – admin



It is always a good practice to change your login credentials

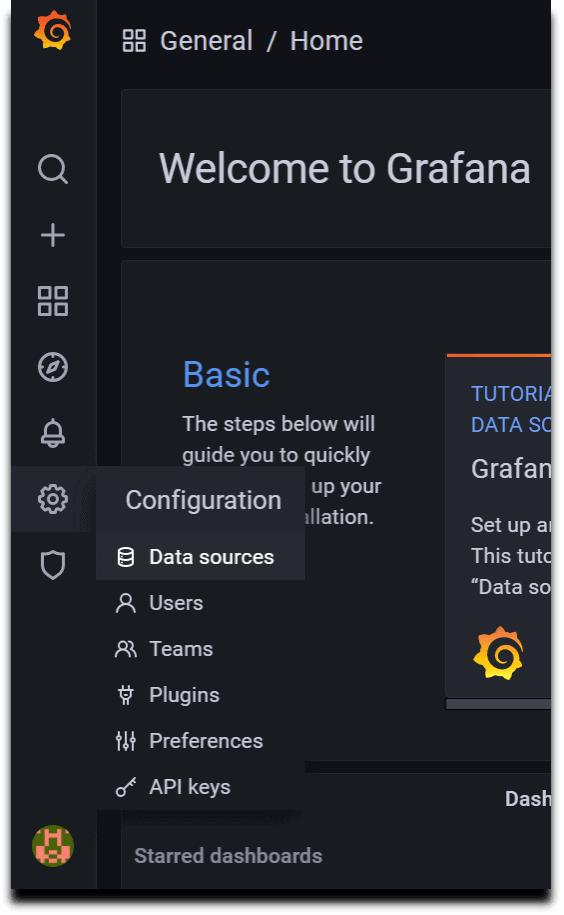


Now here you can see Home Dashboard page of Grafana

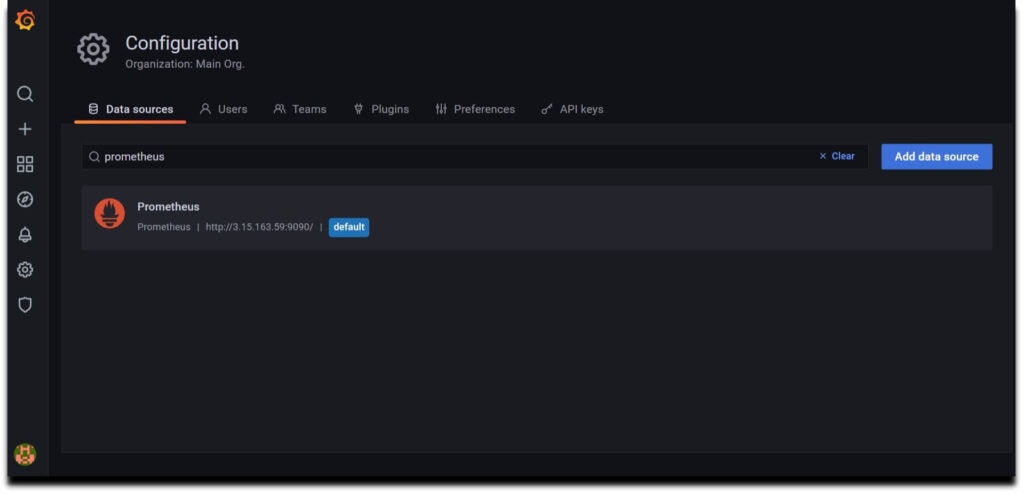


14.Configure Prometheus as Grafana DataSource

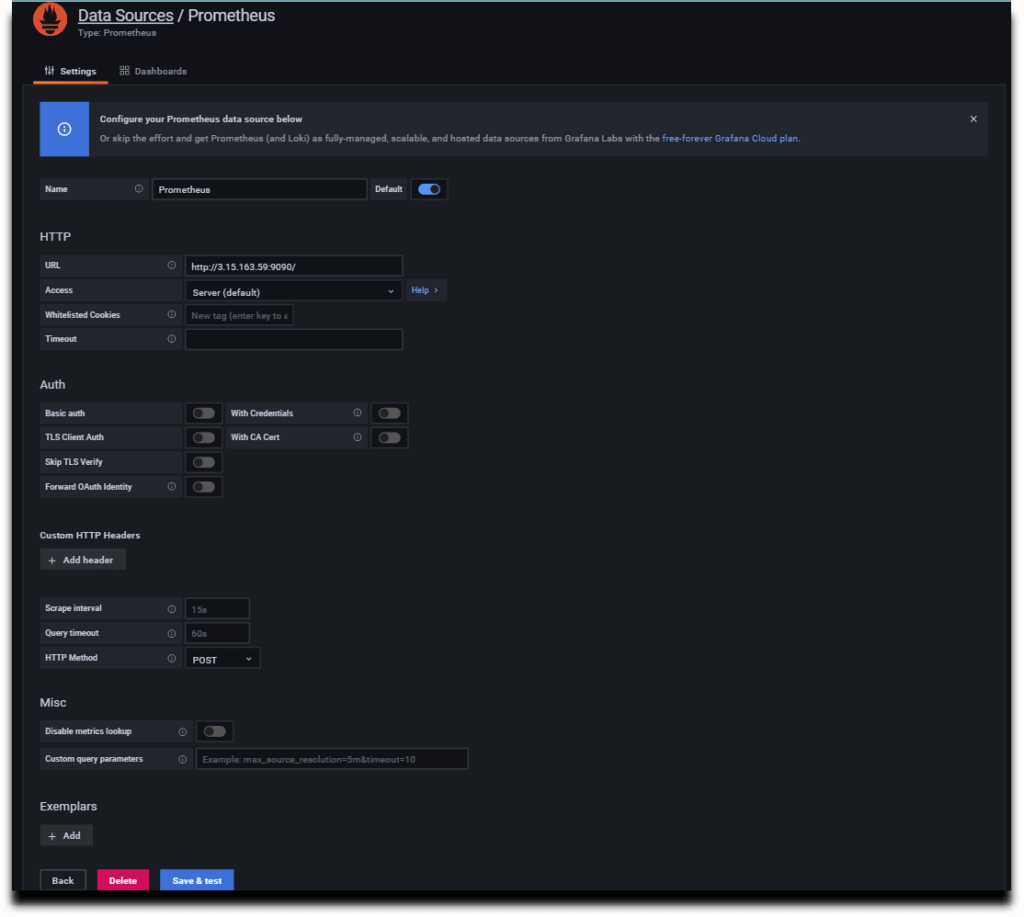
Once you logged into Grafana Now first Navigate to **Settings Icon ->> Configuration ->> data sources**



Now lets click on **Add Data sources**and select **Prometheus**

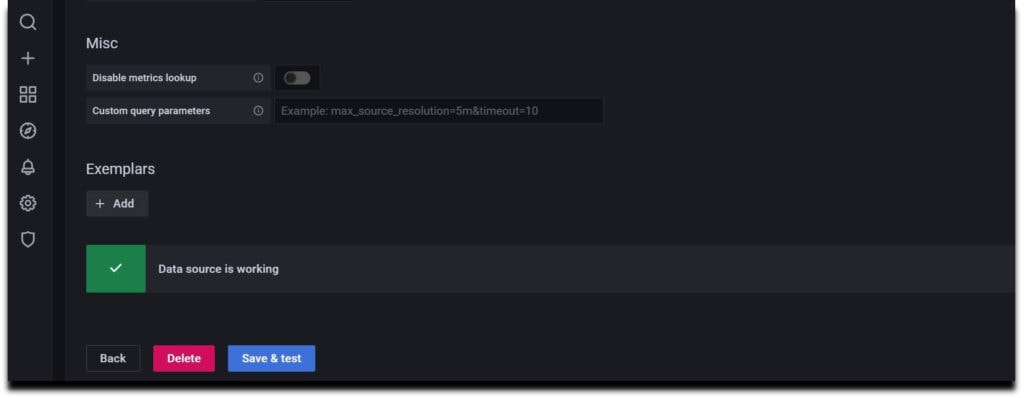


Now configure Prometheus data source by providing Prometheus URL



As per your requirement you can do other changes or you can also keep remaining configuration as default.

Now click on **Save & test**so it will prompt a message**Data Source is working**.



15.Install Node Exporter on Ubuntu

Node Exporter collects the metrics of your system such as Memory usage, CPU usage, RAM, disk space, etc.

# wget <https://github.com/prometheus/node_exporter/releases/download/v1.2.0/node_exporter-1.2.0.linux-amd64.tar.gz>

Unzip the downloaded the file using below command

# sudo tar xvzf node\_exporter-1.2.0.linux-amd64.tar.gz

Now do #ls and your can see node\_exporter binary file.

Go to that file and move this file to your /usr/local/bin directory using below command

# cd node\_exporter-1.2.0.linux-amd64

# sudo cp node\_exporter /usr/local/bin

#16.Creating Node Exporter Systemd service

Now lets create a node\_exporter service in /lib/systemd/ system directory named node\_exporter.service using below commands.

# cd /lib/systemd/system

# sudo vim node\_exporter.service

|  |
| --- |
| [Unit]  Description=Node Exporter  Wants=network-online.target  After=network-online.target  [Service]  Type=simple  User=node\_exporter  Group=node\_exporter  ExecStart=/usr/local/bin/node\_exporter \  — collector.mountstats \  — collector.logind \  — collector.processes \  — collector.ntp \  — collector.systemd \  — collector.tcpstat \  — collector.wifi  Restart=always  RestartSec=10s  [Install]  WantedBy=multi-user.targe |

# sudo systemctl enable node\_exporter

Reload the systemctl daemon, start Node Exporter, and verify its status. The service should be active

# sudo systemctl daemon-reload

# sudo systemctl start node\_exporter

# sudo systemctl status node\_exporter

#17.Configure the Node Exporter as a Prometheus target

Now to scrape the node\_exporter lets instruct the Prometheus by making a minor change in prometheus.yml file

So go to etc/prometheus and open prometheus.yml

# cd /etc/prometheus

# sudo nano prometheus.yml

Now in static\_configs in your configuration file replace the target line with the below one

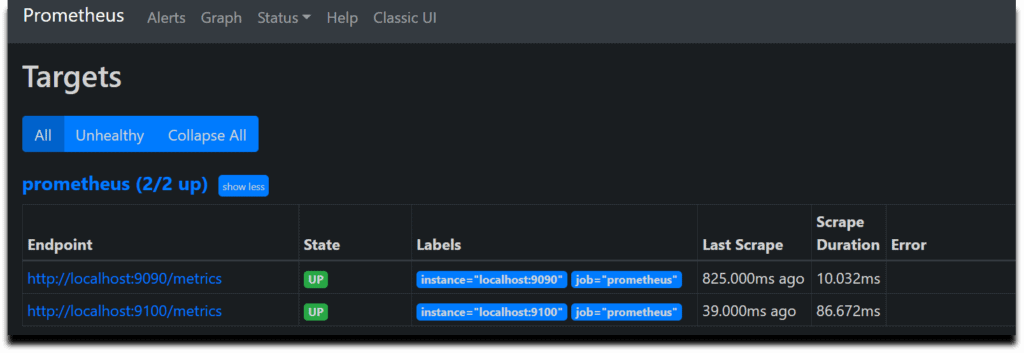
- targets: [‘localhost:9090’, ‘localhost:9100’]

Now restart the Prometheus Service

# sudo systemctl restart prometheus

Hit the URL in your web browser to check weather our target is successfully scraped by Prometheus or not

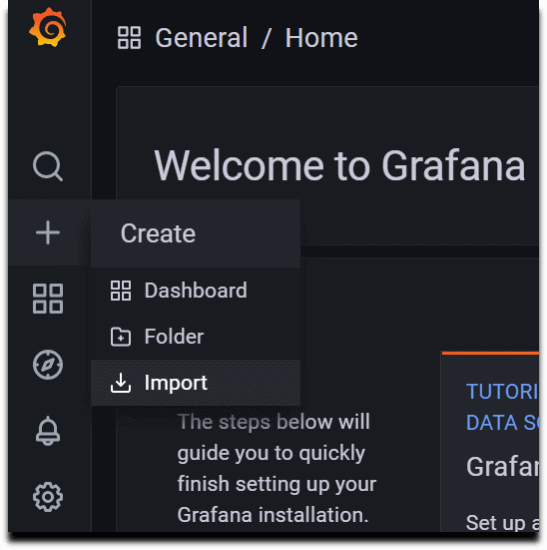
<https://localhost:9100/targets>



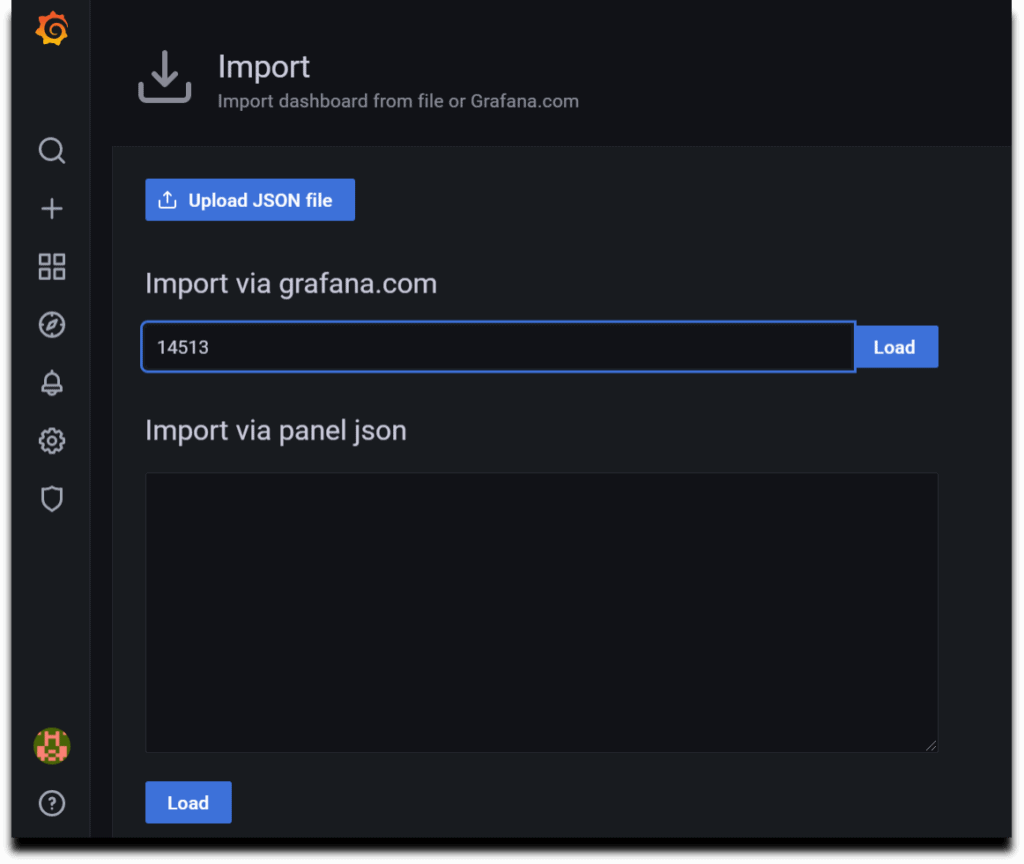
#18.Creating Grafana Dashboard to Monitor Linux Server

Now lets build a dashboard in Grafana so then it will able to reflect the metrics of the Linux system.

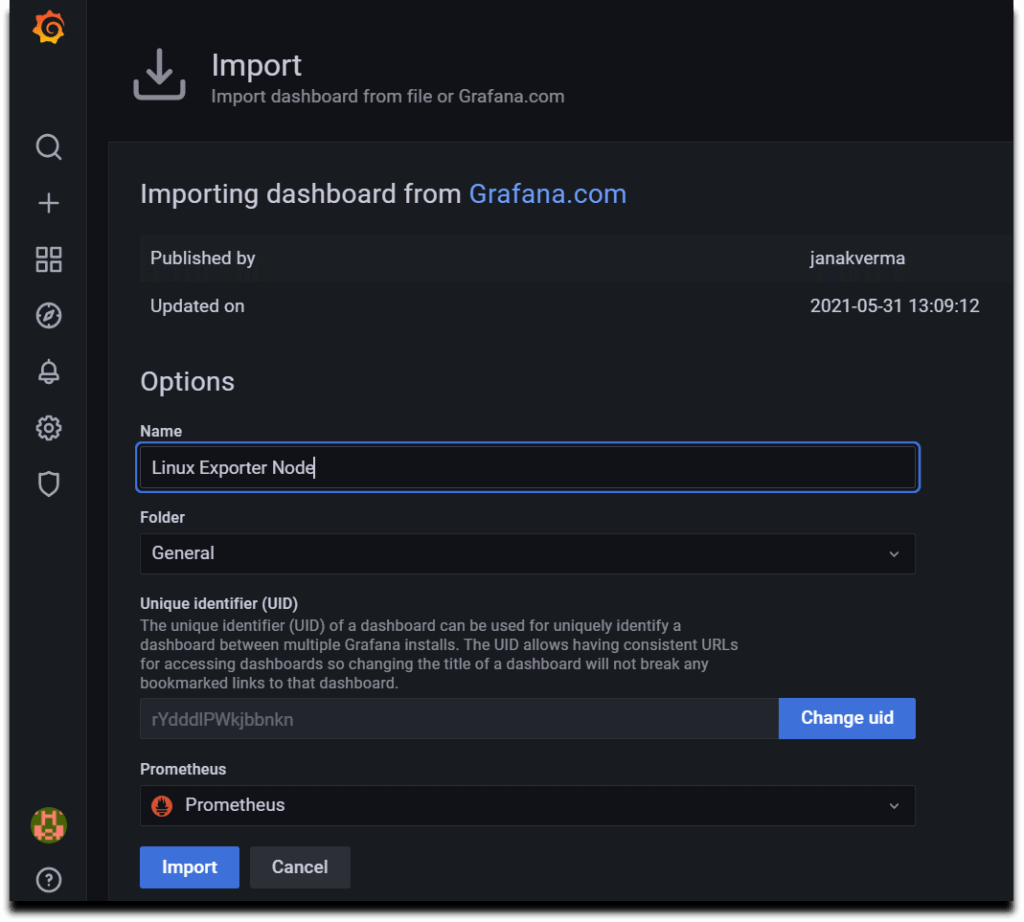
So we will use 14513 to import Grafana.com, Lets come to Grafana Home page and you can see a “+” icon. Click on that and select “Import”



Now provide the Grafana.com Dashboard ID which is 14513 and click on Load



Now provide the name and select the Prometheus Datasource and click on Import.



There you are done with the setup. Now your Dashboard is running up!.

