

1. Install Prometheus Server

- Configuration basic authentication username/password
- Screenshot of login prompt while trying to access prometheus
- Screenshot of prometheus dashboard

We download and unzip the tar file as follows:

```
samana@samana-vm:~$ curl -LO https://github.com/prometheus/prometheus/releases/download/v2.31.1/prometheus-2.31.1.linux-amd64.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100   640    100   640    0     0    760      0  --:--:-- --:--:-- --:--:--    760
100 69.6M   100 69.6M    0     0 3973k      0  0:00:17 0:00:17 --:--:-- 4948k
samana@samana-vm:~$
```

```
samana@samana-vm:~$ tar zxvf prometheus-2.31.1.linux-amd64.tar.gz
prometheus-2.31.1.linux-amd64/
prometheus-2.31.1.linux-amd64/conssoles/
prometheus-2.31.1.linux-amd64/conssoles/index.html.example
prometheus-2.31.1.linux-amd64/conssoles/node-cpu.html
prometheus-2.31.1.linux-amd64/conssoles/node-disk.html
prometheus-2.31.1.linux-amd64/conssoles/node-overview.html
prometheus-2.31.1.linux-amd64/conssoles/node.html
prometheus-2.31.1.linux-amd64/conssoles/prometheus-overview.html
prometheus-2.31.1.linux-amd64/conssoles/prometheus.html
prometheus-2.31.1.linux-amd64/console_libraries/
prometheus-2.31.1.linux-amd64/console_libraries/menu.lib
prometheus-2.31.1.linux-amd64/console_libraries/prom.lib
prometheus-2.31.1.linux-amd64/prometheus.yml
prometheus-2.31.1.linux-amd64/LICENSE
prometheus-2.31.1.linux-amd64/NOTICE
prometheus-2.31.1.linux-amd64/prometheus
prometheus-2.31.1.linux-amd64/promtool
samana@samana-vm:~$
```

Now we create a password with bcrypt module for our prometheus server.

```
samana@samana-vm:~/prometheus-2.31.1.linux-amd64$ python3
Python 3.8.10 (default, Sep 28 2021, 16:10:42)
[GCC 9.3.0] on linux
Type "help", "copyright", "credits" or "license" for more information.
>>> import bcrypt
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
ModuleNotFoundError: No module named 'bcrypt'
>>> import bcrypt
>>> my_pass = bcrypt.hashpw("prompass".encode("utf-8"), bcrypt.gensalt())
>>> my_pass.decode()
'$2b$12$VpCD94mQ1X/rShD2BBE4b0PSAaGUNb3RIyem4QFIH9KgBEtsqTXnG'
>>>
```

```
GNU nano 4.8 web.yml Modified
basic_auth_users:
  samana: $2b$12$VpCD94mQ1X/rSHd2BBE4b0PSAaGUNb3RIyem4QFIH9KgBEtsqTXnG
```

After creating configuration file for user and password we create service file for prometheus as follows:

```
1 [Unit]
2 Description=service for prometheus
3 After=network.target
4 [Service]
5 Type=simple
6 ExecStart=/usr/local/bin/prometheus/prometheus --config.file=/usr/local/bin/prometheus/-
  prometheus.yml --web.config.file=/usr/local/bin/prometheus/web.yml
7
8 #Restart=always
9
10 [Install]
11 WantedBy=multi-user.target
```

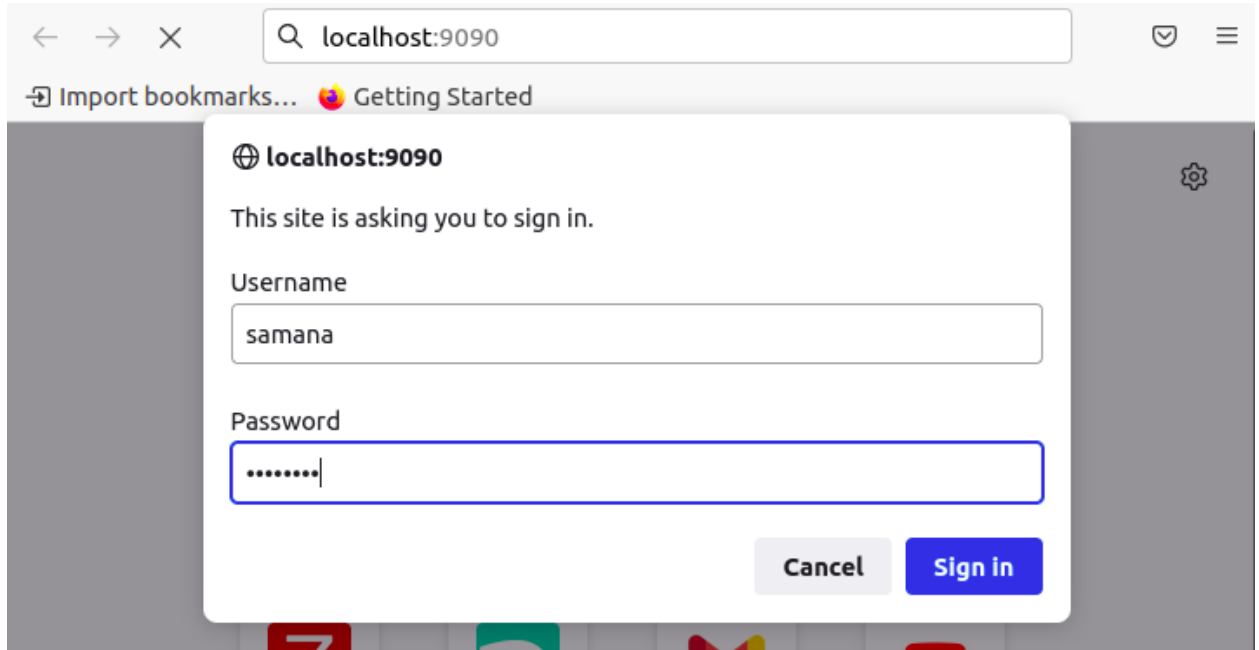
After the service file has been created, we can easily start prometheus as background service. The prometheus service is found to be running as follows:

```
samana@samana-vm:/etc/systemd/system$ sudo systemctl daemon-reload
samana@samana-vm:/etc/systemd/system$ sudo systemctl restart prometheus.service

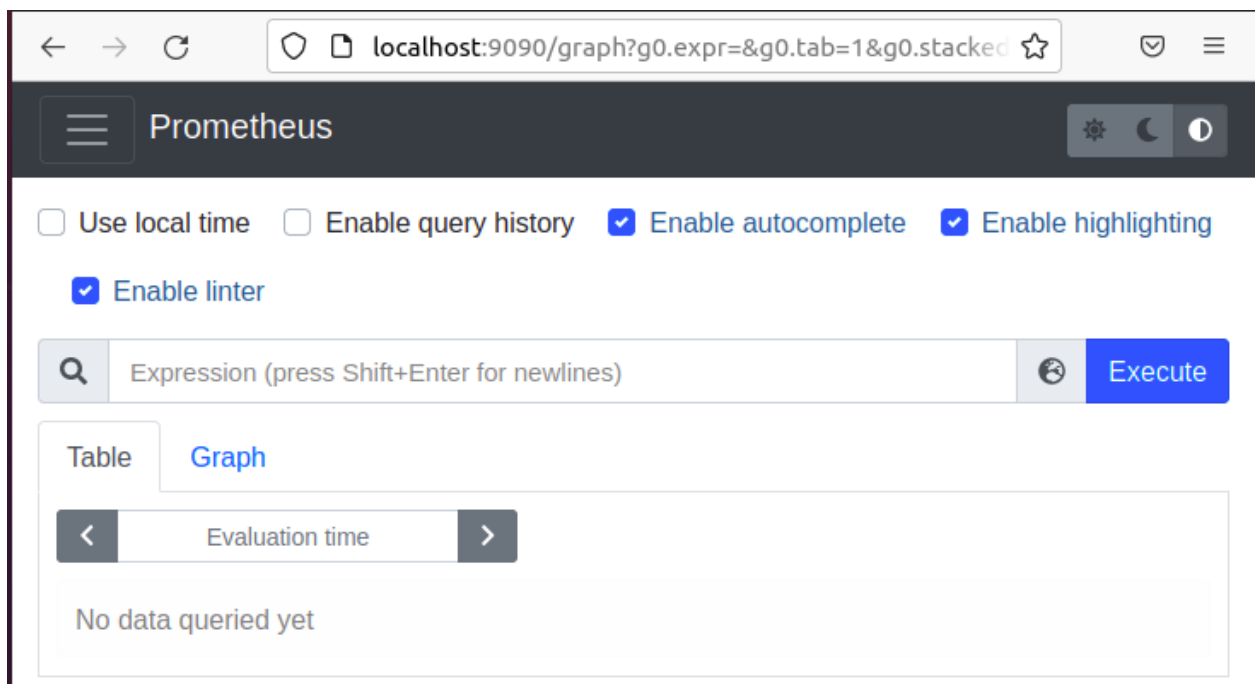
samana@samana-vm:/etc/systemd/system$ sudo systemctl status prometheus.service
● prometheus.service
   Loaded: loaded (/etc/systemd/system/prometheus.service; disabled; vendor >
   Active: active (running) since Sat 2021-12-04 19:19:17 +0545; 10s ago
   Main PID: 4452 (prometheus)
     Tasks: 6 (limit: 3505)
    Memory: 19.1M
    CGroup: /system.slice/prometheus.service
            └─4452 /home/samana/prometheus-2.31.1.linux-amd64/prometheus --co>

दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.419Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.419Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.419Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.419Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.421Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.421Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.422Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.423Z cal>
दि सम्बर 04 19:19:17 samana-vm prometheus[4452]: ts=2021-12-04T13:34:17.423Z cal>
दि सम्बर 04 19:19:27 samana-vm systemd[1]: /etc/systemd/system/prometheus.servic>
samana@samana-vm:/etc/systemd/system$
```

Login prompt while trying to access prometheus: basic auth



Prometheus dashboard:



2. Install node exporter on another machine than the server
 - Add that machine target to server configuration
 - Share screenshot from status->targets to show the available nodes
 - Share configuration of node exporter & prometheus server

We extract the downloaded tar file for node_exporter as follows:

```
samana@samana:~/Downloads$ sudo tar -zxvf node_exporter-1.3.0.linux-amd64.tar.gz
node_exporter-1.3.0.linux-amd64/
node_exporter-1.3.0.linux-amd64/LICENSE
node_exporter-1.3.0.linux-amd64/NOTICE
node_exporter-1.3.0.linux-amd64/node_exporter
samana@samana:~/Downloads$
```

```
samana@samana:~/Downloads$ sudo cp node_exporter-1.3.0.linux-amd64/node_exporter /usr/local/bin
samana@samana:~/Downloads$ sudo chown node_exporter:node_exporter /usr/local/bin/node_exporter
samana@samana:~/Downloads$
```

We create a service file for node_exporter as follows

```
Open  ▼  [+]
```

***node_exporter.service**
/etc/systemd/system

Save ≡ _ □

```
1 [Unit]
2 Description=Node Exporter
3 Wants=network-online.target
4 After=network-online.target
5
6 [Service]
7 User=node_exporter
8 Group=node_exporter
9 Type=simple
10 ExecStart=/usr/local/bin/node_exporter
11
12 [Install]
13 WantedBy=multi-user.target
14
```

We start the node_exporter service as follows:

```
samana@samana:~/Downloads$ sudo systemctl daemon-reload
samana@samana:~/Downloads$ sudo systemctl start node_exporter.service
samana@samana:~/Downloads$ sudo systemctl status node_exporter.service
● node_exporter.service - Node Exporter
   Loaded: loaded (/etc/systemd/system/node_exporter.service; disabled;>
   Active: active (running) since Sun 2021-12-05 09:42:11 +0545; 9s ago
     Main PID: 7057 (node_exporter)
        Tasks: 5 (limit: 14131)
       Memory: 2.3M
      CGroup: /system.slice/node_exporter.service
             └─7057 /usr/local/bin/node_exporter

दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.998>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.998>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.998>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.998>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.998>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.998>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.998>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.999>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.999>
दि सम्बर 05 09:42:12 samana node_exporter[7057]: ts=2021-12-05T03:57:11.999>
lines 1-19/19 (END)
```

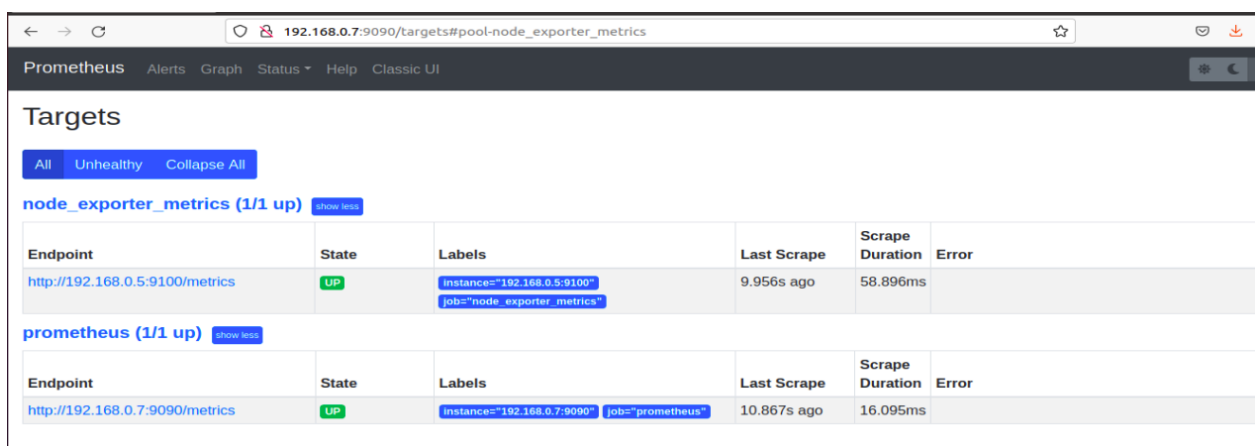


Now in order to add node exporter to target in our prometheus server we edit the configuration file as follows:

```
prometheus.yml
/usr/local/bin/prometheus

3 scrape_interval: 15s # Set the scrape interval to every 15 seconds. Default is every 1
  minute.
4 evaluation_interval: 15s # Evaluate rules every 15 seconds. The default is every 1 minute.
5 # scrape_timeout is set to the global default (10s).
6
7 # Alertmanager configuration
8 alerting:
9   alertmanagers:
10     - static_configs:
11       - targets:
12         # - alertmanager:9093
13
14 # Load rules once and periodically evaluate them according to the global 'evaluation_interval'.
15 rule_files:
16   # - "first_rules.yml"
17   # - "second_rules.yml"
18
19 # A scrape configuration containing exactly one endpoint to scrape:
20 # Here it's Prometheus itself.
21 scrape_configs:
22   # The job name is added as a label `job=<job_name>` to any timeseries scraped from this
  config.
23   - job_name: "prometheus"
24
25     # metrics_path defaults to '/metrics'
26     # scheme defaults to 'http'.
27
28     static_configs:
29       - targets: ["192.168.0.7:9090"]
30     basic_auth:
31       username: samana
32       password: prompass
33   - job_name: 'node_exporter_metrics'
34     # If prometheus-node-exporter is installed, grab stats about the local
35     scrape_interval: 5s
36     # machine by default.
37     static_configs:
38       - targets: ["192.168.0.5:9100"]
```

We can see that the node_exporter has been created and is up as a target in our prometheus dashboard.



The screenshot shows the Prometheus web interface at 192.168.0.7:9090/targets#pool-node_exporter_metrics. The 'Targets' section is active, showing two target groups. The first group, 'node_exporter_metrics (1/1 up)', has one target at 'http://192.168.0.5:9100/metrics' which is 'UP'. The second group, 'prometheus (1/1 up)', has one target at 'http://192.168.0.7:9090/metrics' which is also 'UP'.

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.0.5:9100/metrics	UP	instance="192.168.0.5:9100" job="node_exporter_metrics"	9.956s ago	58.896ms	

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.0.7:9090/metrics	UP	instance="192.168.0.7:9090" job="prometheus"	10.867s ago	16.095ms	

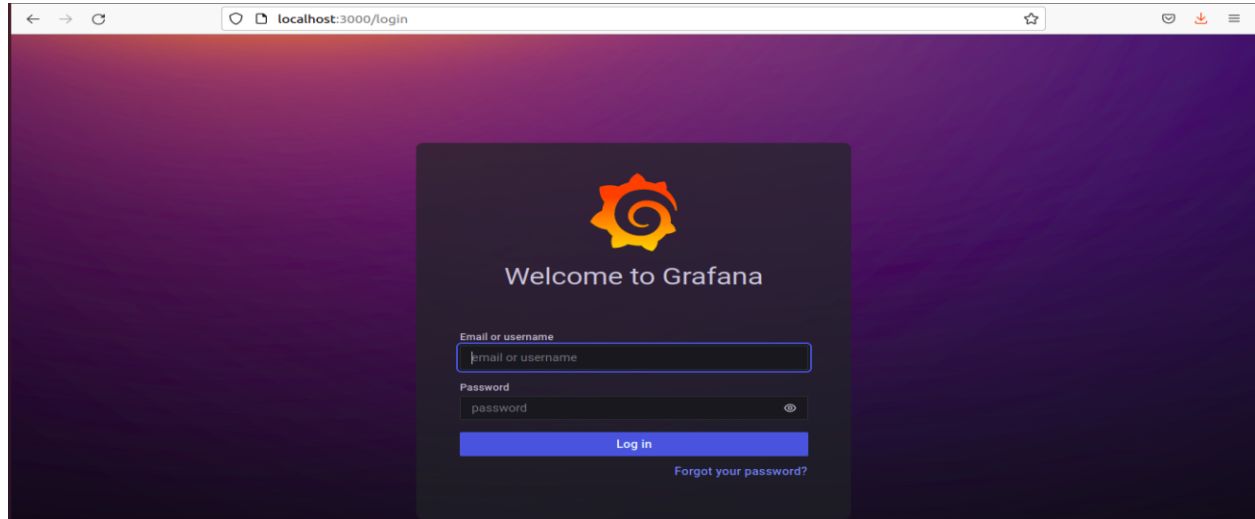
- Add prometheus data source to grafana, should be connected through basic auth
- Screenshot of working data source config
- Import & apply dashboard for node_exporter
- Screenshot of dashboard of nodes with live metrics show

```
samana@samana-vm:~$ wget https://dl.grafana.com/enterprise/release/grafana-enterprise_8.3.0_amd64.deb
--2021-12-05 12:29:26-- https://dl.grafana.com/enterprise/release/grafana-enterprise_8.3.0_amd64.deb
Resolving dl.grafana.com (dl.grafana.com)... 151.101.198.217, 2a04:4e42:a::729
Connecting to dl.grafana.com (dl.grafana.com)|151.101.198.217|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 82223034 (78M) [application/vnd.debian.binary-package]
Saving to: 'grafana-enterprise_8.3.0_amd64.deb'

md64.deb                50%[=====>                ] 39.21M  4.67MB/s   eta 10s

samana@samana-vm:~$ sudo dpkg -i grafana-enterprise_8.3.0_amd64.deb
Selecting previously unselected package grafana-enterprise.
(Reading database ... 188015 files and directories currently installed.)
Preparing to unpack grafana-enterprise_8.3.0_amd64.deb ...
Unpacking grafana-enterprise (8.3.0) ...
```

[illegible]



We edit the prometheus.yml file to add the grafana as a target in prometheus server dashboard.

```
Open  *prometheus.yml  Save  ~/Desktop
19 # A scrape configuration containing exactly one endpoint to scrape.
20 # Here it's Prometheus itself.
21 scrape_configs:
22   # The job name is added as a label `job=<job_name>` to any timeseries scraped from this
  config.
23   - job_name: "prometheus"
24
25     # metrics_path defaults to '/metrics'
26     # scheme defaults to 'http'.
27
28     static_configs:
29       - targets: ["192.168.0.7:9090"]
30   basic_auth:
31     username: samana
32     password: prompass
33   - job_name: 'node_exporter_metrics'
34     # If prometheus-node-exporter is installed, grab stats about the local
35     scrape_interval: 5s
36     # machine by default.
37     static_configs:
38       - targets: ["192.168.0.5:9100"]
39   - job_name: 'grafana'
40
41     scrape_interval: 5s
42     static_configs:
43       - targets: ["192.168.0.7:3000"]
```


← → ↻ 192.168.0.7:9090/targets#pool-grafana ☆ 🌙 📄

Prometheus Alerts Graph Status ▾ Help Classic UI ⚙️ 🌙 📄

Targets

All Unhealthy Collapse All

grafana (1/1 up) [show less](#)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.0.7:3000/metrics	UP	instance="192.168.0.7:3000" job="grafana"	574.000ms ago	9.174ms	

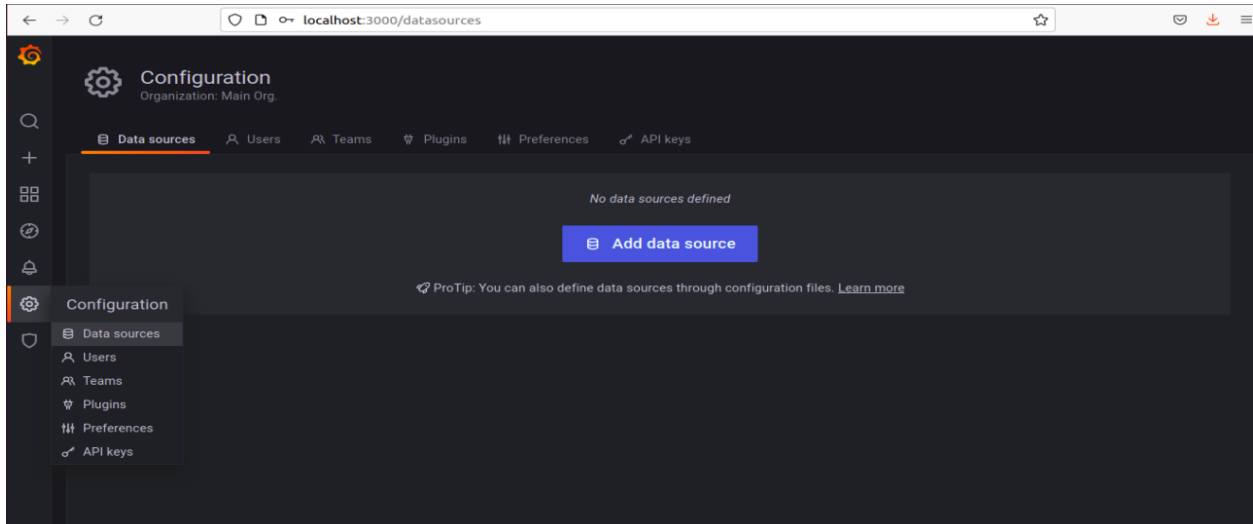
node_exporter (1/1 up) [show less](#)

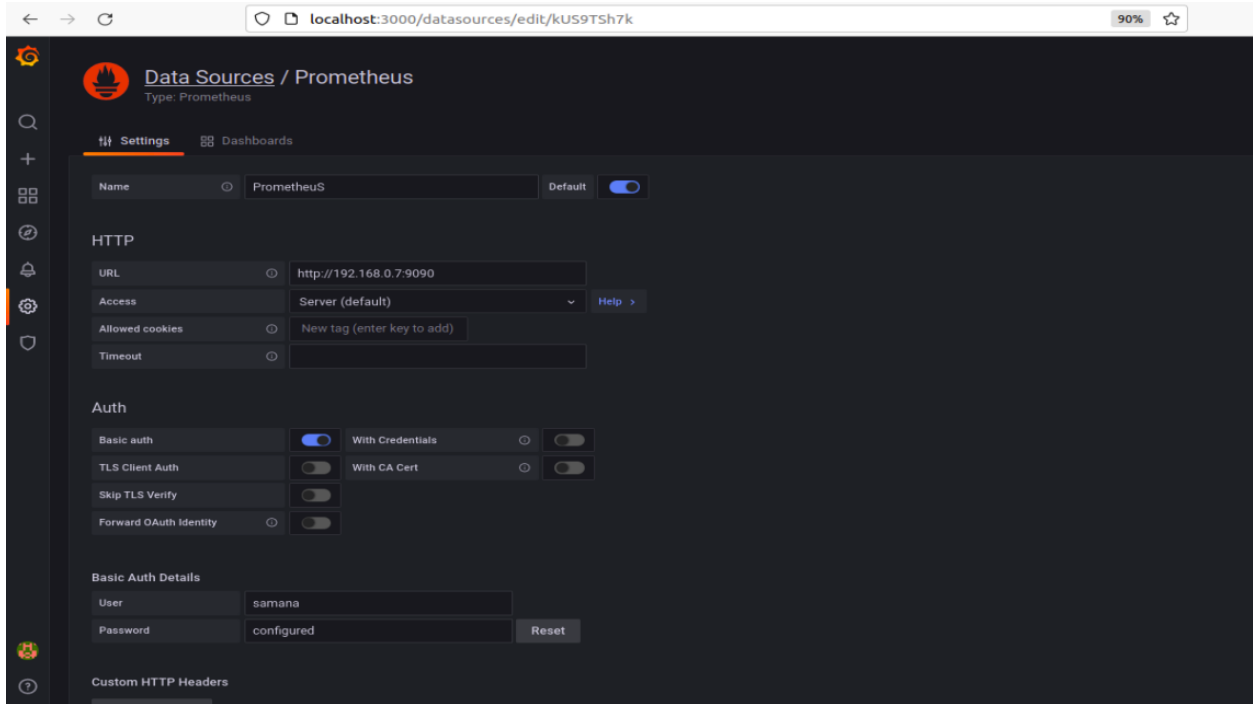
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.0.5:9100/metrics	UP	instance="192.168.0.5:9100" job="node_exporter"	427.000ms ago	58.569ms	

prometheus (1/1 up) [show less](#)

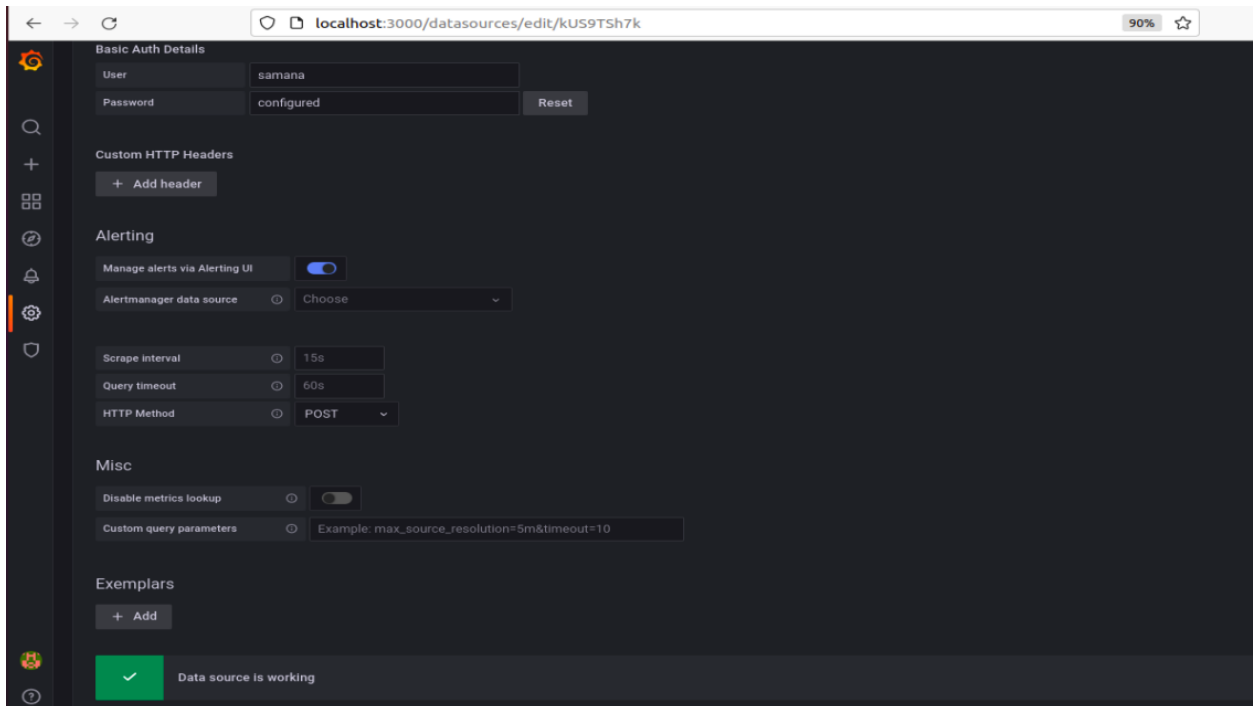
Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://192.168.0.7:9090/metrics	UP	instance="192.168.0.7:9090" job="prometheus"	2.695s ago	9.522ms	

Now in order to add prometheus data source with basic auth in grafana, we perform the following actions

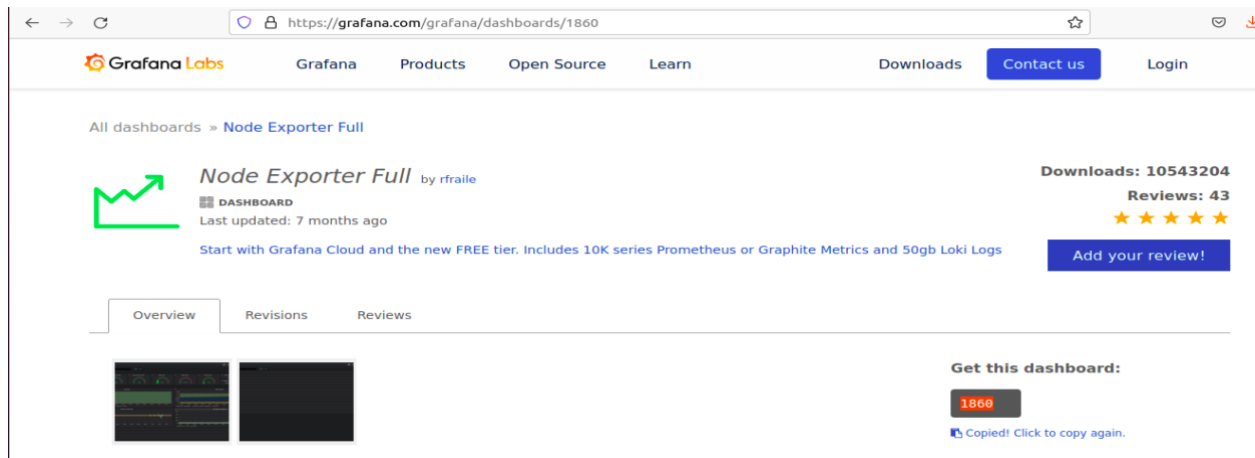
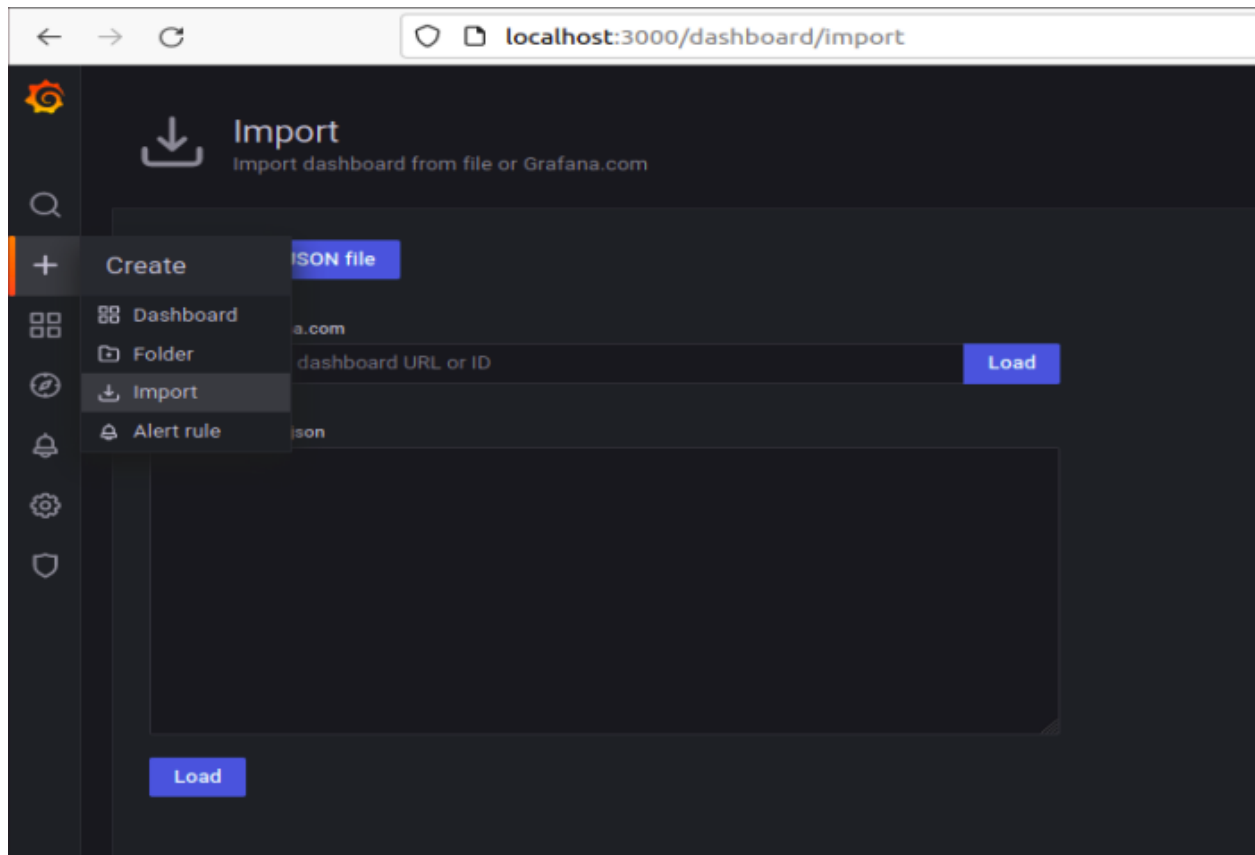


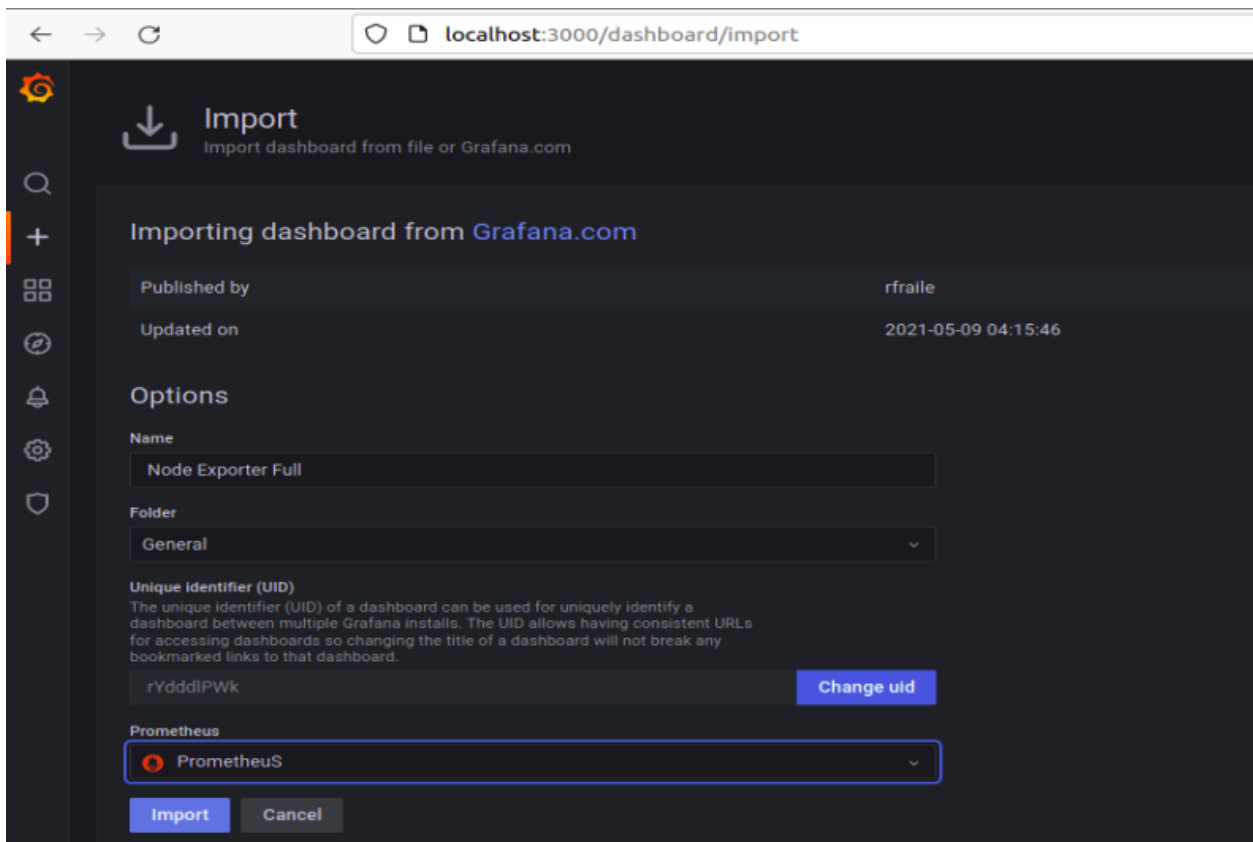
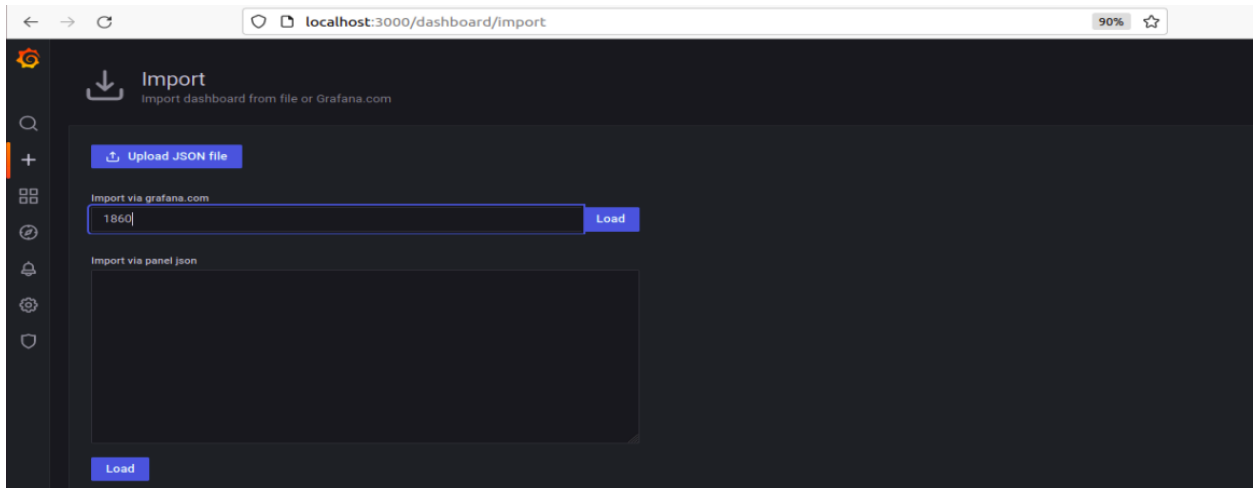


After applying this configuration, we can see that the data source is working properly.



Now, we import the dashboard for node exporter as follows:





dashboard of node_exporter with live metrics:

