Lesson 02 Demo 02

Writing Basic Queries in PromQL

Objective: To query and analyze monitoring data using Prometheus Query Language (PromQL) for effective monitoring of system performance and health

Tools required: Linux operating system

Prerequisites: Refer to Demo 01 of Lesson 02 for configuring Node Exporter

Steps to be followed:

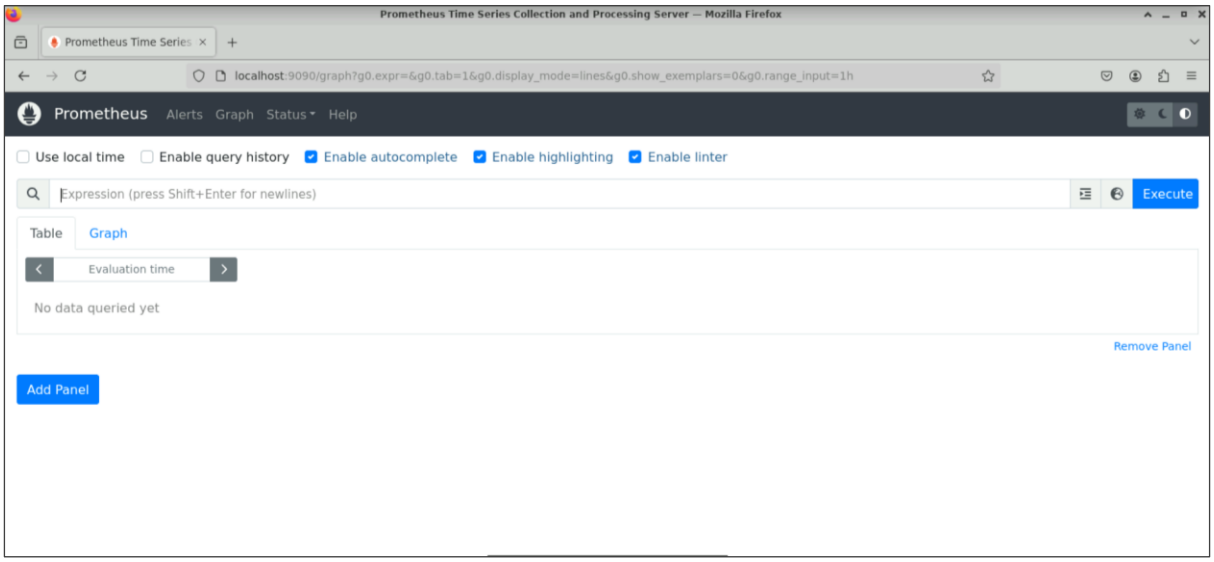
1. Query to retrieve a single metric
2. Filter by label
3. Aggregate data with the sum() function
4. Query data using an arithmetic operation
5. Calculate a metric using the rate() function

Step 1: Query to retrieve a single metric

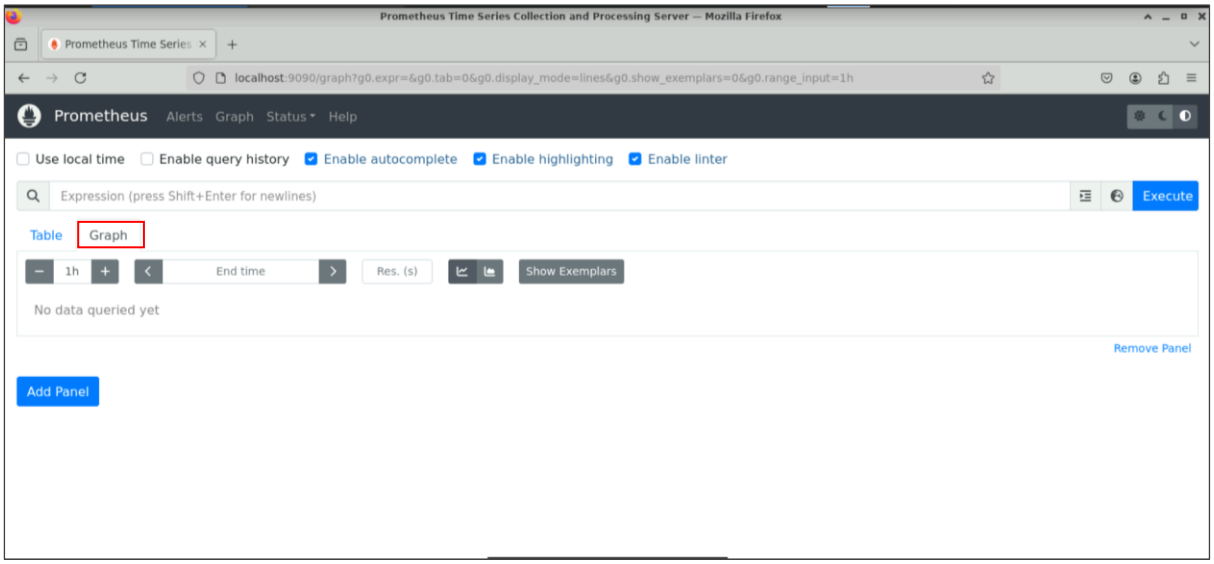
1.1 Navigate to the terminal and run the following command to start the Prometheus server:
sudo /usr/local/bin/prometheus --config.file=/etc/prometheus/prometheus.yml

```
labuser@ip-172-31-24-36:~$ sudo /usr/local/bin/prometheus --config.file=/etc/prometheus/prometheus.yml
ts=2025-01-29T07:06:16.907Z caller=main.go:601 level=info msg="No time or size retention was set so using the default time retention" duration=15d
ts=2025-01-29T07:06:16.907Z caller=main.go:645 level=info msg="Starting Prometheus Server" mode=server version="(version=2.54.0, branch=HEAD, revisio
n=5354e87a70d3eb26b81b601b286d66ff983990f6)"
ts=2025-01-29T07:06:16.907Z caller=main.go:650 level=info build_context="(go=go1.22.6, platform=linux/arm64, user=root@68a9e2472a68, date=20240809-11
:38:48, tags=netgo,builtinassets,stringlabels)"
ts=2025-01-29T07:06:16.907Z caller=main.go:651 level=info host_details="(Linux 6.5.0-1017-aws #17~22.04.2-Ubuntu SMP Mon Mar 25 20:47:22 UTC 2024 aar
ch64 ip-172-31-24-36 (none))"
ts=2025-01-29T07:06:16.907Z caller=main.go:652 level=info fd_limits="(soft=1048576, hard=1048576)"
ts=2025-01-29T07:06:16.907Z caller=main.go:653 level=info vm_limits="(soft=unlimited, hard=unlimited)"
ts=2025-01-29T07:06:16.921Z caller=web.go:571 level=info component=web msg="Start listening for connections" address=0.0.0.0:9090
ts=2025-01-29T07:06:16.925Z caller=main.go:1160 level=info msg="Starting TSDB ..."
ts=2025-01-29T07:06:16.928Z caller=tls_config.go:313 level=info component=web msg="Listening on" address=[::]:9090
ts=2025-01-29T07:06:16.929Z caller=tls_config.go:316 level=info component=web msg="TLS is disabled." http2=false address=[::]:9090
ts=2025-01-29T07:06:16.956Z caller=head.go:626 level=info component=tsdb msg="Replaying on-disk memory mappable chunks if any"
ts=2025-01-29T07:06:16.962Z caller=head.go:713 level=info component=tsdb msg="On-disk memory mappable chunks replay completed" duration=31.795µs
ts=2025-01-29T07:06:16.962Z caller=head.go:721 level=info component=tsdb msg="Replaying WAL, this may take a while"
ts=2025-01-29T07:06:16.964Z caller=head.go:793 level=info component=tsdb msg="WAL segment loaded" segment=0 maxSegment=0
ts=2025-01-29T07:06:16.964Z caller=head.go:830 level=info component=tsdb msg="WAL replay completed" checkpoint_replay_duration=71.926µs wal_replay_du
ration=1.364563ms wbl_replay_duration=206ns chunk_snapshot_load_duration=0s mmap_chunk_replay_duration=31.795µs total_replay_duration=1.529067ms
ts=2025-01-29T07:06:16.967Z caller=main.go:1181 level=info fs_type=EXT4_SUPER_MAGIC
ts=2025-01-29T07:06:16.967Z caller=main.go:1184 level=info msg="TSDB started"
```

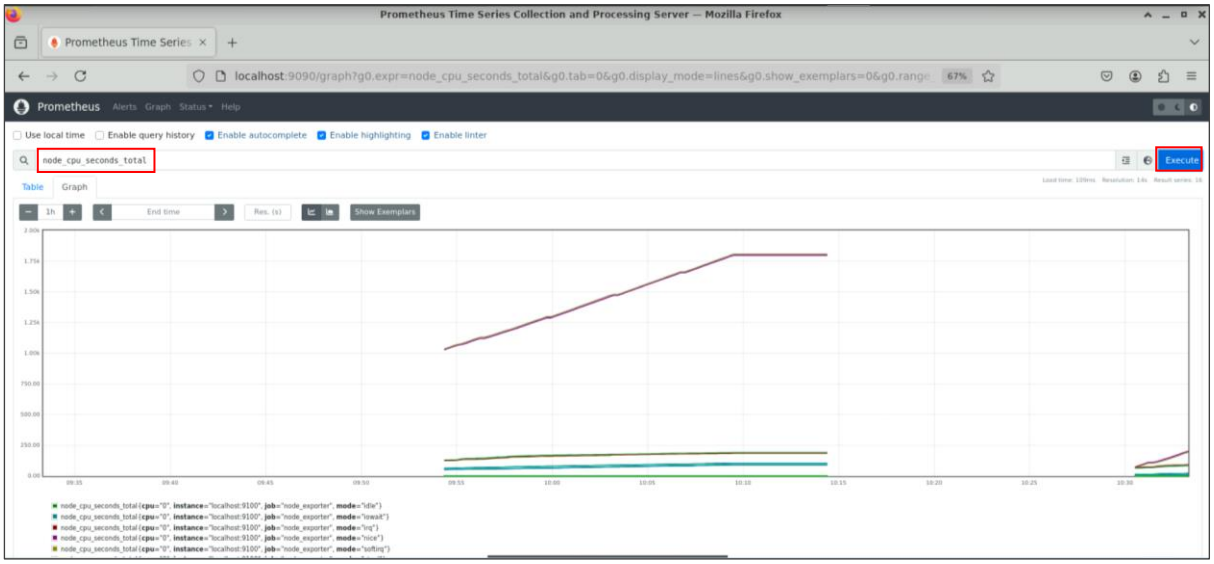
1.2 Navigate to the browser and enter the URL **http://localhost:9090/** or **http://<public-ip>:9090/** to access the Prometheus console



1.3 Navigate to the **Graph** section

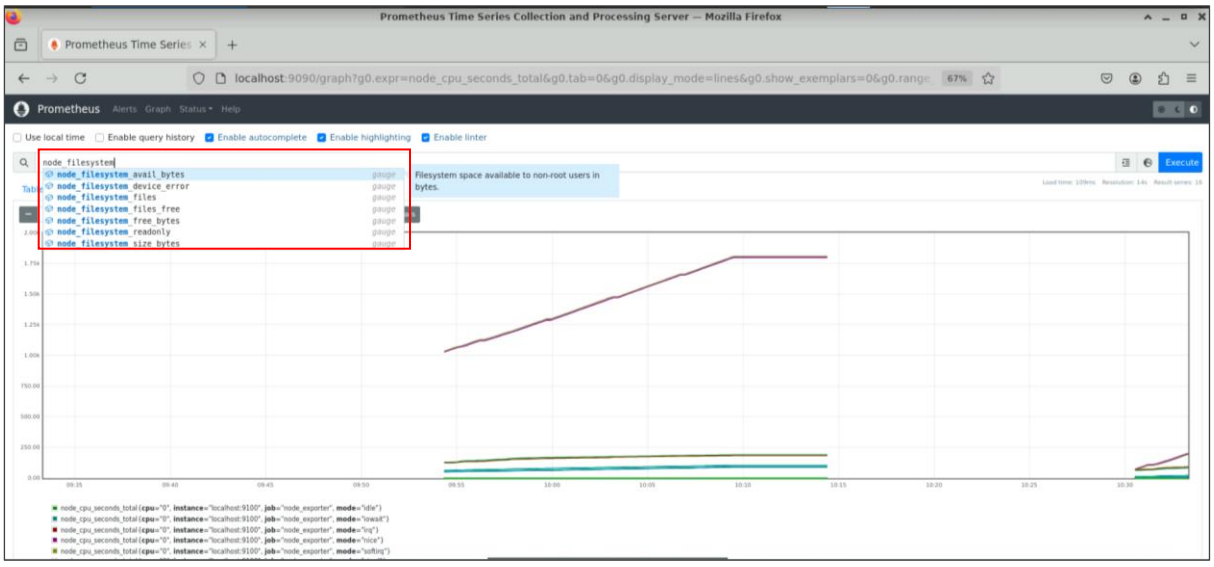


1.4 Enter the following query in the expression browser to retrieve a single metric, then click on **Execute**:
node_cpu_seconds_total



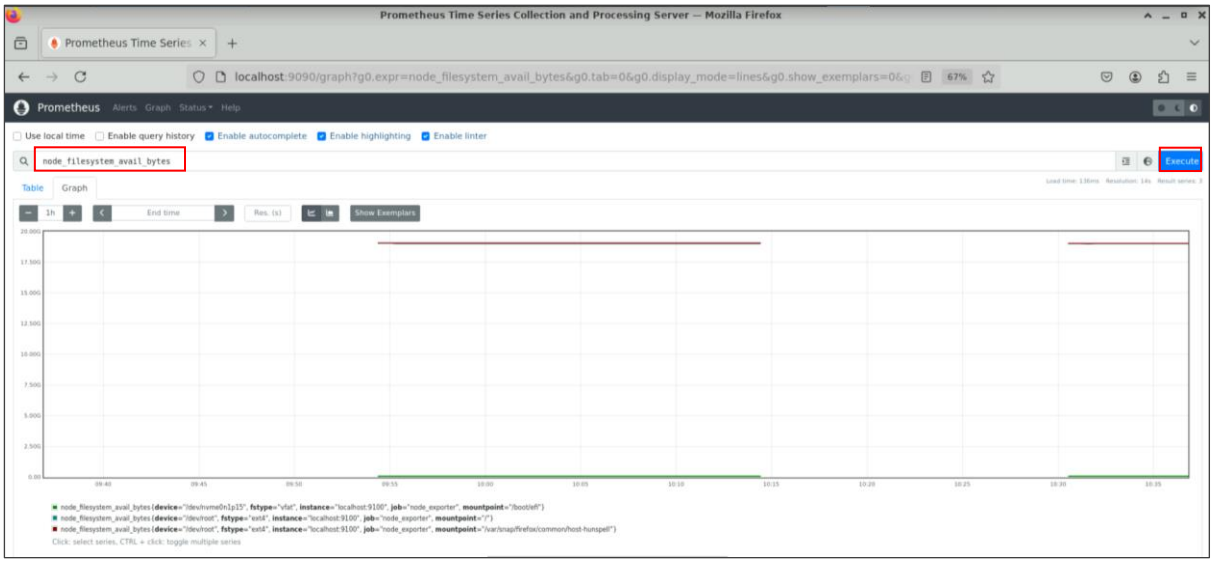
Step 2: Filter by label

2.1 Type **node_filesystem** in the expression browser

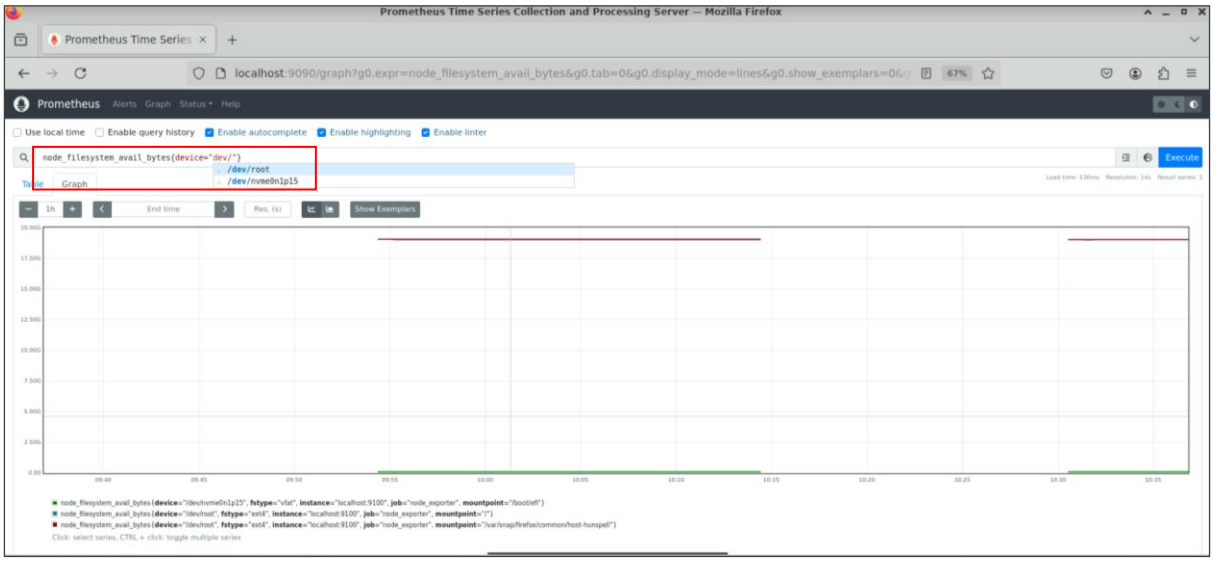


It will display a popup list.

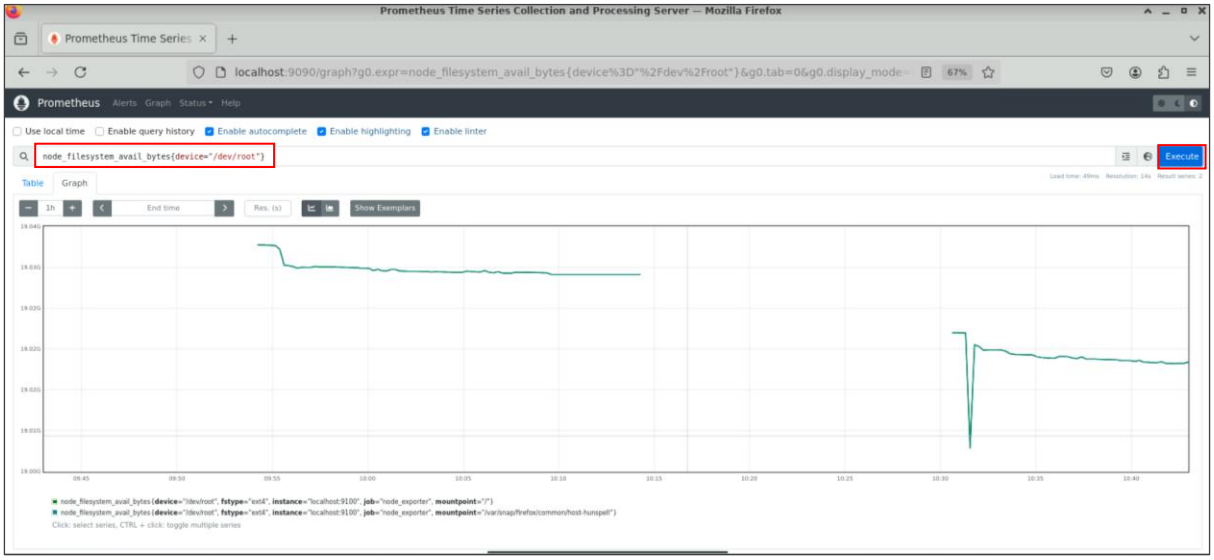
2.2 Select **node_filesystem_avail_bytes** and click **Execute**



2.3 Filter by labels using the following query:
node_filesystem_avail_bytes{device=”/dev/”}

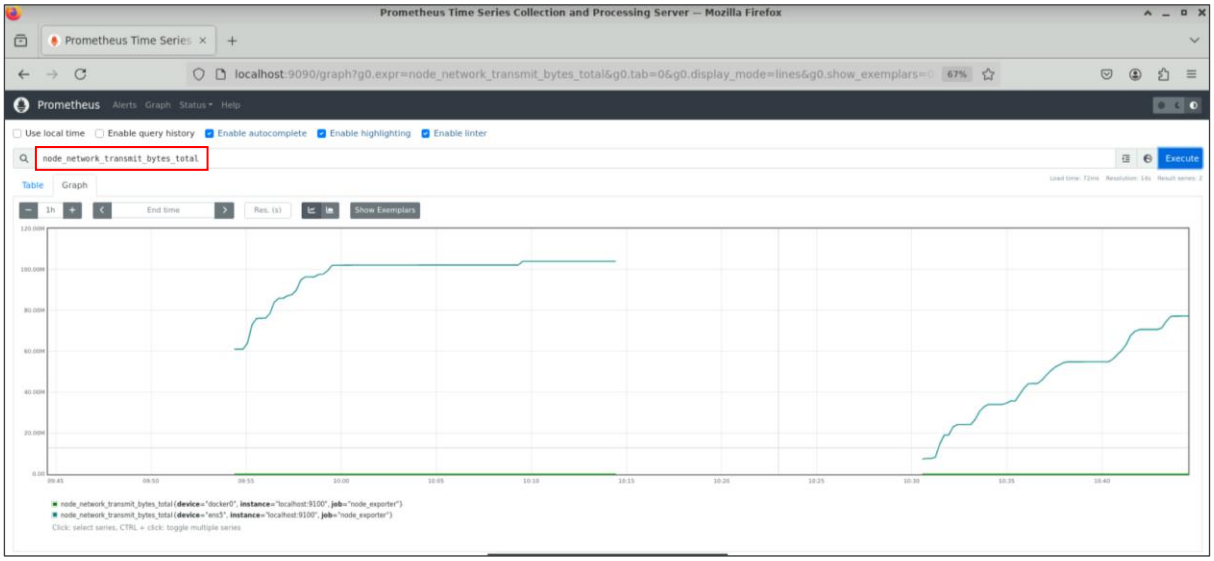


2.4 Select **/dev/root** and click on **Execute** to run the query

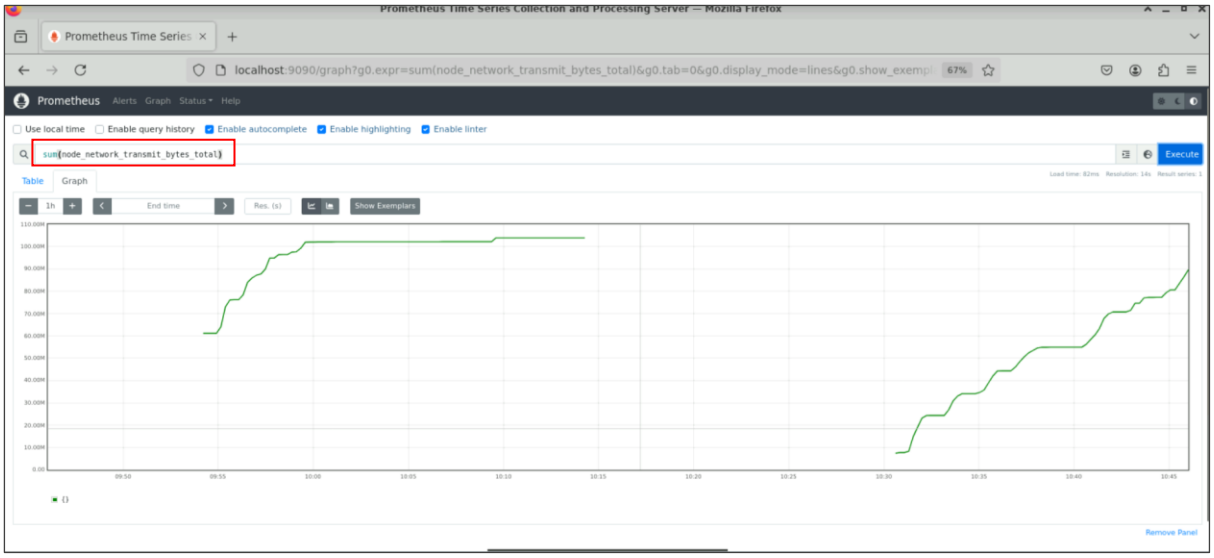


Step 3: Aggregate data with the sum() function

3.1 In the expression browser, enter the following query and execute it:
node_network_transmit_bytes_total

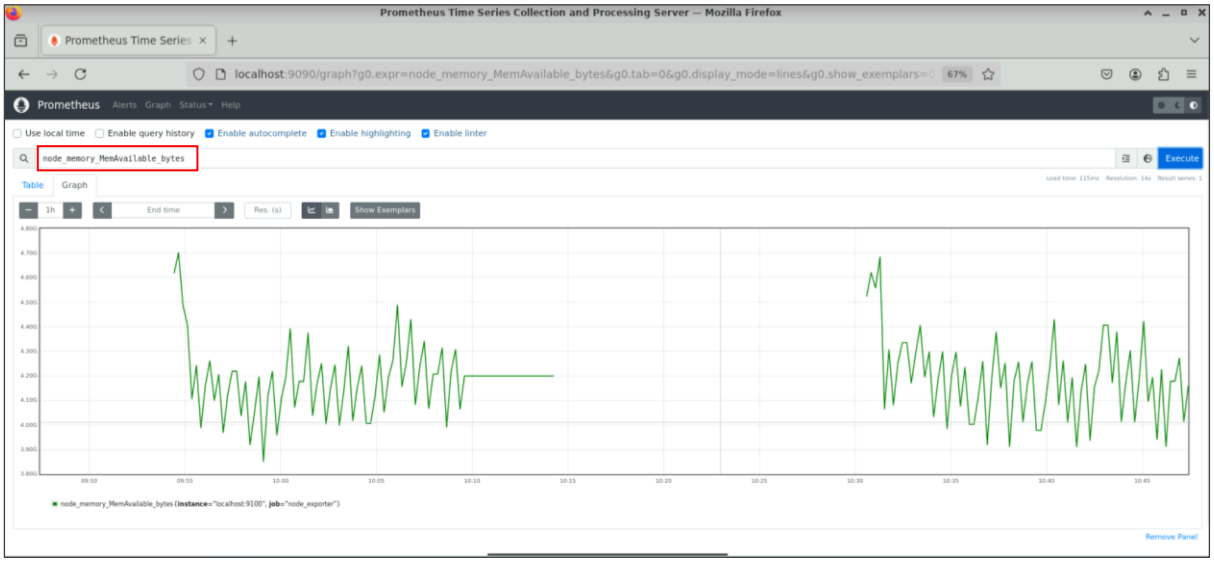


3.2 Use the following query to sum the total number of bytes transmitted over the network interface and view the graph:
sum(node_network_transmit_bytes_total)

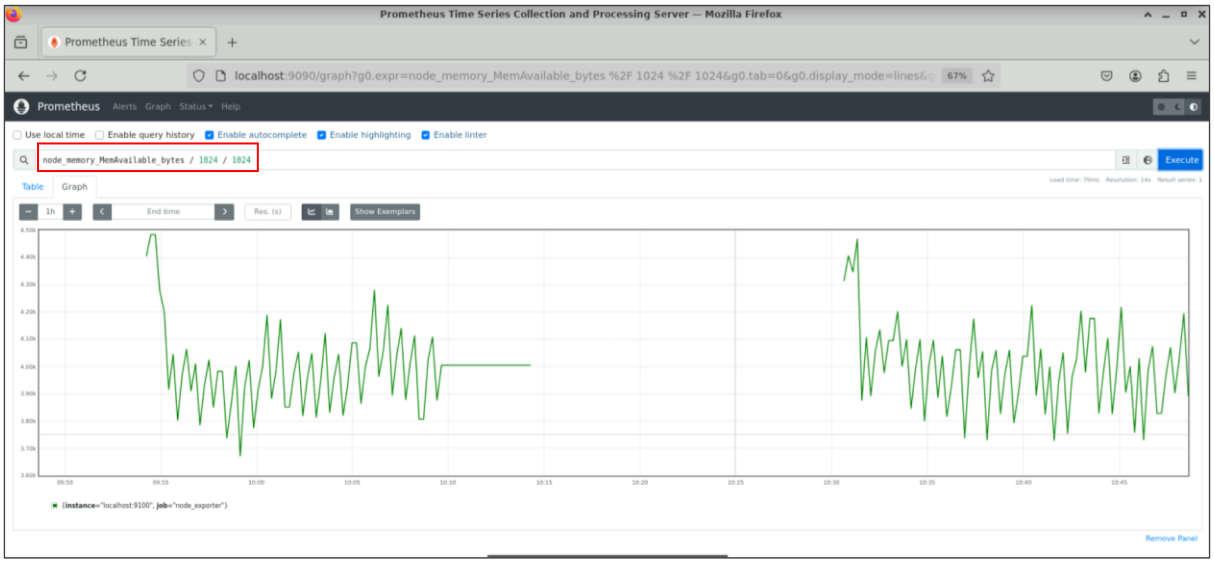


Step 4: Query data using an arithmetic operation

4.1 Enter the following query to display the amount of available memory on a node in bytes:
node_memory_MemAvailable_bytes

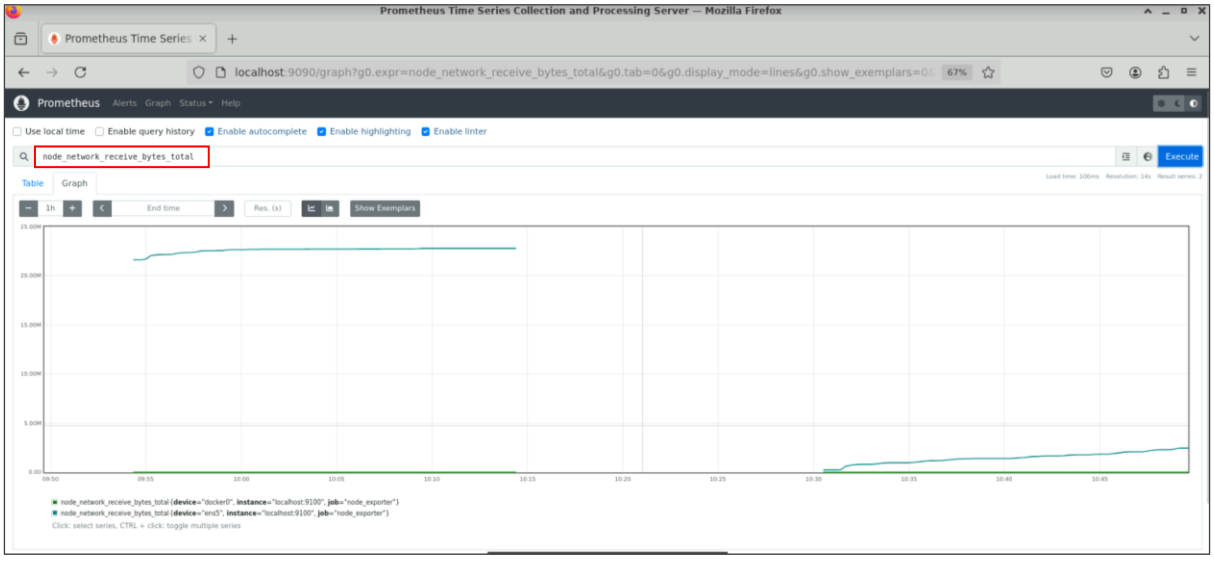


4.2 Divide the query executed in the previous step by **1024** twice to display it in megabytes using the following query:
node_memory_MemAvailable_bytes / 1024 / 1024

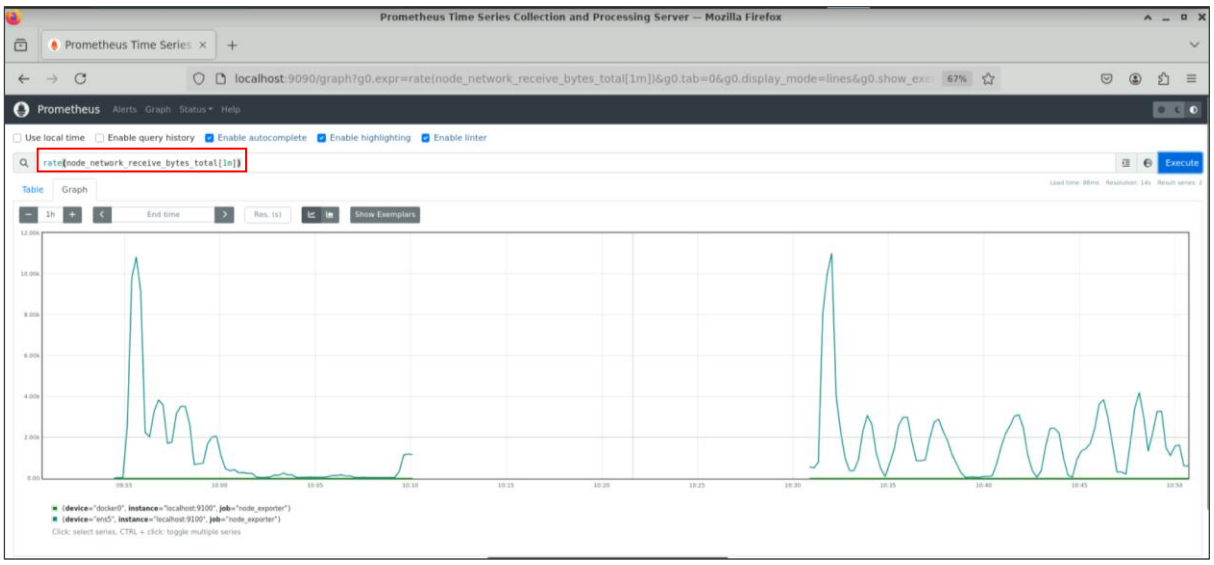


Step 5: Calculate a metric using the rate() function

5.1 Execute the following query in the expression browser to represent the total number of bytes received over the network interface since the system started:
node_network_receive_bytes_total



5.2 Enter the following query to calculate the average per-second rate of bytes received over the network interface in the last minute:
rate(node_network_receive_bytes_total[1m])



By following these steps, you have successfully queried and analyzed monitoring data using Prometheus Query Language to effectively monitor system performance and health.