Lesson End Project

Monitoring Apache Server Metrics Using Prometheus

Project agenda: To set up Prometheus to monitor an Apache web server, collect metrics using Apache Exporter, and analyze performance using PromQL queries

Description: As a DevOps engineer at a company running a high-traffic e-commerce website with Apache web servers, your team has identified performance issues during peak hours. Your task is to implement a monitoring solution to identify potential bottlenecks and ensure optimal performance. This involves configuring Prometheus to monitor the Apache web servers and collect relevant metrics. You are responsible for setting up Apache Exporter and Prometheus and using PromQL queries to track metrics. This setup will help proactively identify and address performance issues.

Tools required: Linux operating system, Prometheus, and Apache web server

Prerequisites: Refer to Demo 02 of Lesson 01 for installing a Prometheus server

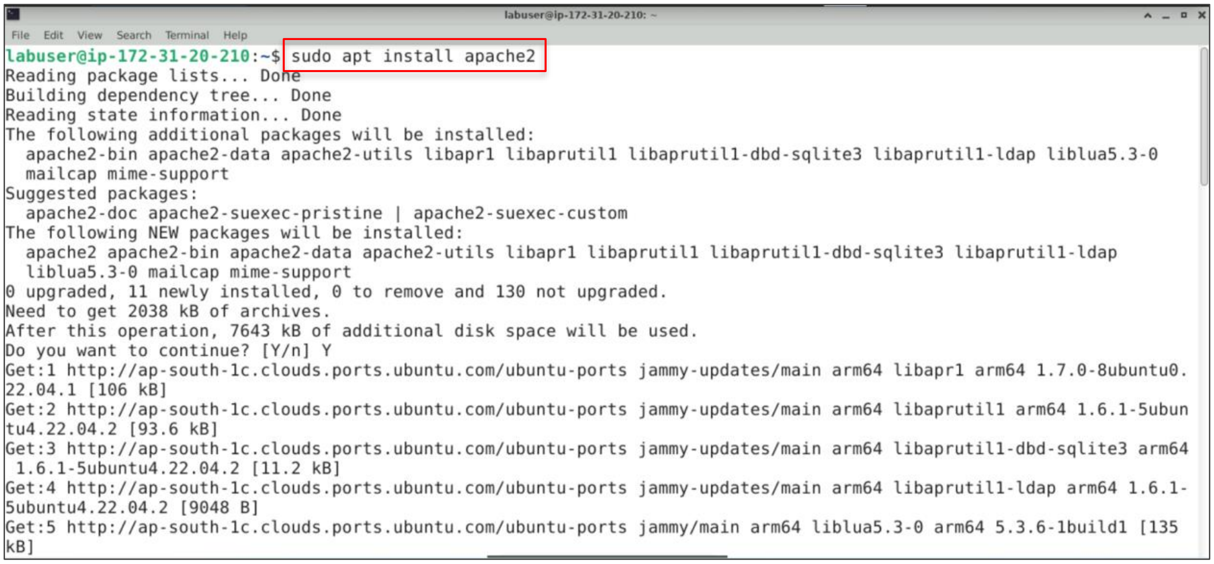
Expected deliverables: A fully configured Prometheus and Apache Exporter setup to collect and visualize Apache web server metrics, accessible through a web interface for real-time monitoring and performance analysis using custom PromQL queries

Steps to be followed:

1. Install and configure the Apache web server
2. Install and configure the Apache Exporter for Prometheus
3. Configure Prometheus to scrape metrics from the Apache Exporter
4. Run PromQL queries in the Prometheus UI to analyze Apache web server metrics

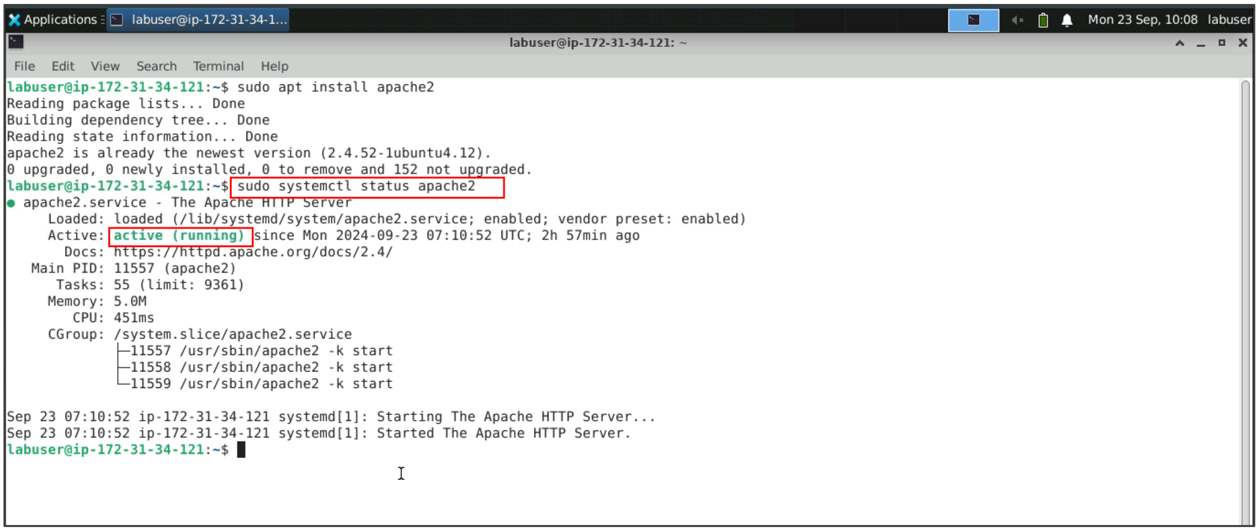
Step 1: Install and configure the Apache web server

1.1 Run the following command in the terminal to install the Apache2 server:
sudo apt install apache2


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.3-0
  mailcap mime-support
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap
  liblua5.3-0 mailcap mime-support
0 upgraded, 11 newly installed, 0 to remove and 130 not upgraded.
Need to get 2038 kB of archives.
After this operation, 7643 kB of additional disk space will be used.
Do you want to continue? [Y/n] Y
Get:1 http://ap-south-1c.clouds.ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 libapr1 arm64 1.7.0-8ubuntu0.
22.04.1 [106 kB]
Get:2 http://ap-south-1c.clouds.ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 libaprutil1 arm64 1.6.1-5ubun
tu4.22.04.2 [93.6 kB]
Get:3 http://ap-south-1c.clouds.ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 libaprutil1-dbd-sqlite3 arm64
 1.6.1-5ubuntu4.22.04.2 [11.2 kB]
Get:4 http://ap-south-1c.clouds.ports.ubuntu.com/ubuntu-ports jammy-updates/main arm64 libaprutil1-ldap arm64 1.6.1-
5ubuntu4.22.04.2 [9048 B]
Get:5 http://ap-south-1c.clouds.ports.ubuntu.com/ubuntu-ports jammy/main arm64 liblua5.3-0 arm64 5.3.6-1build1 [135
kB]
```

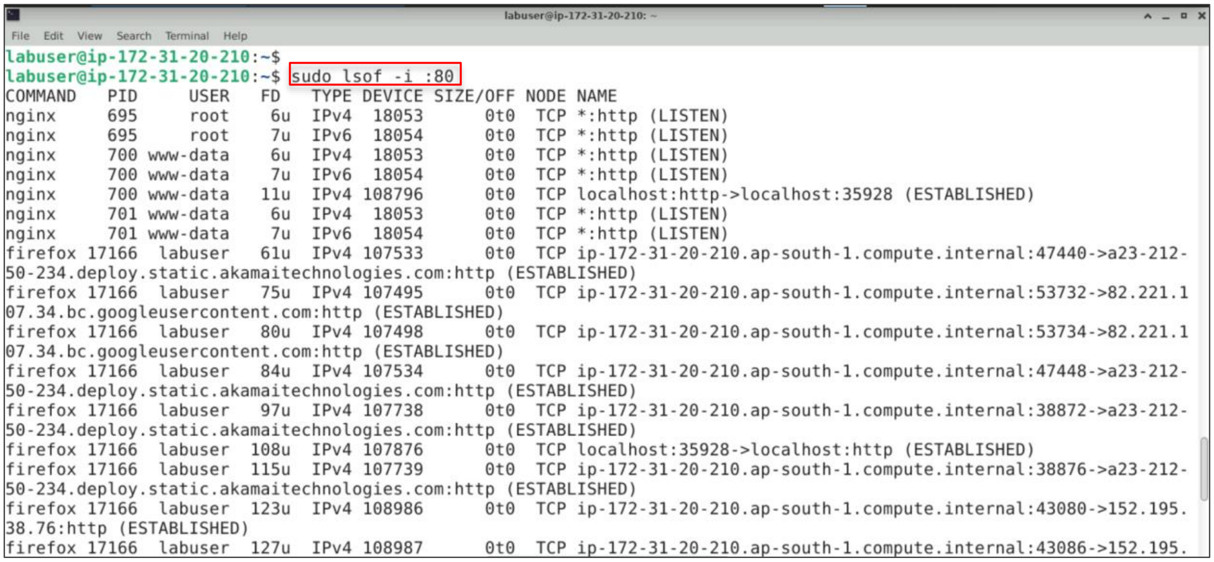
1.2 Run the following command to check the status of the Apache2 service:
sudo systemctl status apache2


Applications
labuser@ip-172-31-34-1...
Mon 23 Sep, 10:08 labuser
labuser@ip-172-31-34-121: ~
File Edit View Search Terminal Help

```
labuser@ip-172-31-34-121:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
apache2 is already the newest version (2.4.52-1ubuntu4.12).
0 upgraded, 0 newly installed, 0 to remove and 152 not upgraded.
labuser@ip-172-31-34-121:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2024-09-23 07:10:52 UTC; 2h 57min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 11557 (apache2)
      Tasks: 55 (limit: 9361)
     Memory: 5.0M
        CPU: 451ms
     CGroup: /system.slice/apache2.service
             ├─11557 /usr/sbin/apache2 -k start
             ├─11558 /usr/sbin/apache2 -k start
             └─11559 /usr/sbin/apache2 -k start

Sep 23 07:10:52 ip-172-31-34-121 systemd[1]: Starting The Apache HTTP Server...
Sep 23 07:10:52 ip-172-31-34-121 systemd[1]: Started The Apache HTTP Server.
labuser@ip-172-31-34-121:~$
```

1.3 Run the following command to check the services that are occupying port 80 if the Apache2 server is not running:

sudo lsof -i :80


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

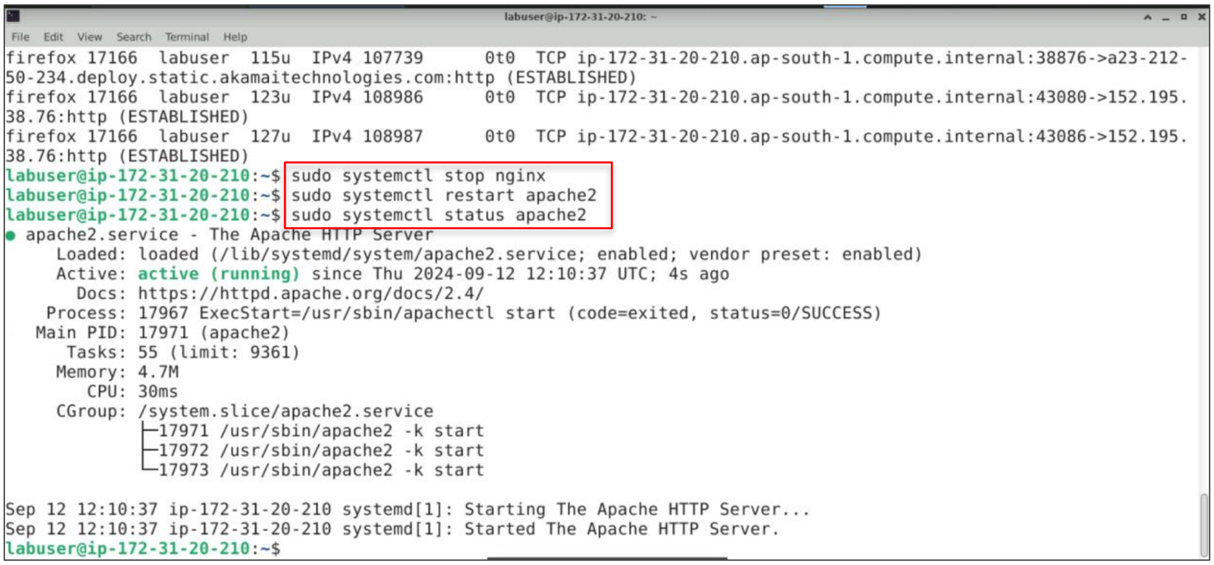
```
labuser@ip-172-31-20-210:~$
labuser@ip-172-31-20-210:~$ sudo lsof -i :80
COMMAND   PID     USER   FD   TYPE DEVICE SIZE/OFF NODE NAME
nginx     695     root    6u  IPv4  18053      0t0  TCP *:http (LISTEN)
nginx     695     root    7u  IPv6  18054      0t0  TCP *:http (LISTEN)
nginx     700 www-data    6u  IPv4  18053      0t0  TCP *:http (LISTEN)
nginx     700 www-data    7u  IPv6  18054      0t0  TCP *:http (LISTEN)
nginx     700 www-data   11u  IPv4 108796      0t0  TCP localhost:http->localhost:35928 (ESTABLISHED)
nginx     701 www-data    6u  IPv4  18053      0t0  TCP *:http (LISTEN)
nginx     701 www-data    7u  IPv6  18054      0t0  TCP *:http (LISTEN)
firefox 17166  labuser   61u  IPv4 107533      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:47440->a23-212-
50-234.deploy.static.akamaitechnologies.com:http (ESTABLISHED)
firefox 17166  labuser   75u  IPv4 107495      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:53732->82.221.1
07.34.bc.googleusercontent.com:http (ESTABLISHED)
firefox 17166  labuser   80u  IPv4 107498      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:53734->82.221.1
07.34.bc.googleusercontent.com:http (ESTABLISHED)
firefox 17166  labuser   84u  IPv4 107534      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:47448->a23-212-
50-234.deploy.static.akamaitechnologies.com:http (ESTABLISHED)
firefox 17166  labuser   97u  IPv4 107738      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:38872->a23-212-
50-234.deploy.static.akamaitechnologies.com:http (ESTABLISHED)
firefox 17166  labuser  108u  IPv4 107876      0t0  TCP localhost:35928->localhost:http (ESTABLISHED)
firefox 17166  labuser  115u  IPv4 107739      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:38876->a23-212-
50-234.deploy.static.akamaitechnologies.com:http (ESTABLISHED)
firefox 17166  labuser  123u  IPv4 108986      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:43080->152.195.
38.76:http (ESTABLISHED)
firefox 17166  labuser  127u  IPv4 108987      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:43086->152.195.
```

1.4 Run the following commands to stop the Nginx service, restart the Apache2 server, and check the status of the Apache2 service:

sudo systemctl stop nginx

sudo systemctl restart apache2

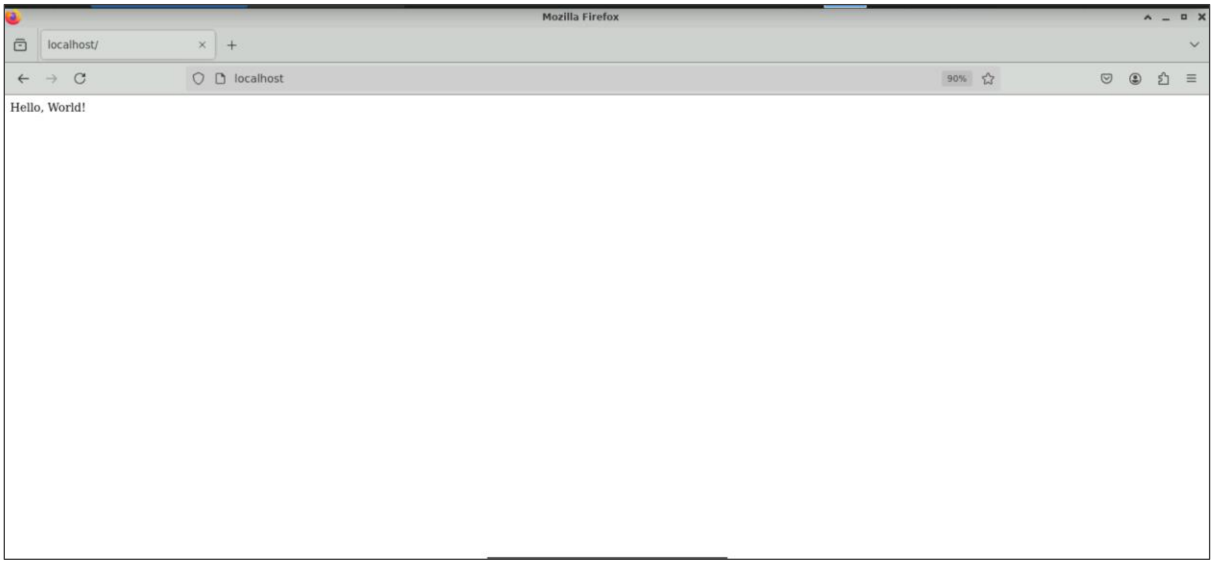
sudo systemctl status apache2


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
firefox 17166  labuser  115u  IPv4 107739      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:38876->a23-212-
50-234.deploy.static.akamaitechnologies.com:http (ESTABLISHED)
firefox 17166  labuser  123u  IPv4 108986      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:43080->152.195.
38.76:http (ESTABLISHED)
firefox 17166  labuser  127u  IPv4 108987      0t0  TCP ip-172-31-20-210.ap-south-1.compute.internal:43086->152.195.
38.76:http (ESTABLISHED)
labuser@ip-172-31-20-210:~$ sudo systemctl stop nginx
labuser@ip-172-31-20-210:~$ sudo systemctl restart apache2
labuser@ip-172-31-20-210:~$ sudo systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2024-09-12 12:10:37 UTC; 4s ago
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 17967 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 17971 (apache2)
      Tasks: 55 (limit: 9361)
     Memory: 4.7M
        CPU: 30ms
     CGroup: /system.slice/apache2.service
             ├─17971 /usr/sbin/apache2 -k start
             ├─17972 /usr/sbin/apache2 -k start
             └─17973 /usr/sbin/apache2 -k start

Sep 12 12:10:37 ip-172-31-20-210 systemd[1]: Starting The Apache HTTP Server...
Sep 12 12:10:37 ip-172-31-20-210 systemd[1]: Started The Apache HTTP Server.
labuser@ip-172-31-20-210:~$
```

Note: Check ports **80** for HTTP and **443** for HTTPS. In this case, it is **Nginx**, but it could be any process. Stop the process occupying these ports to start Apache2

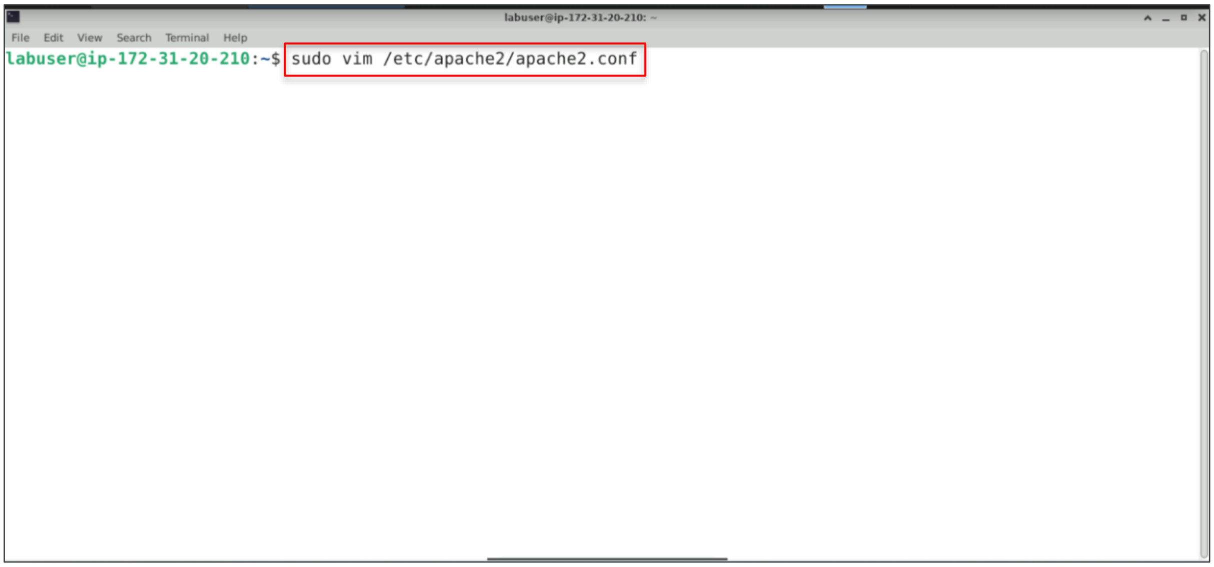
1.5 Navigate to the preferred browser and enter the URL **http://localhost**



Hello, World!

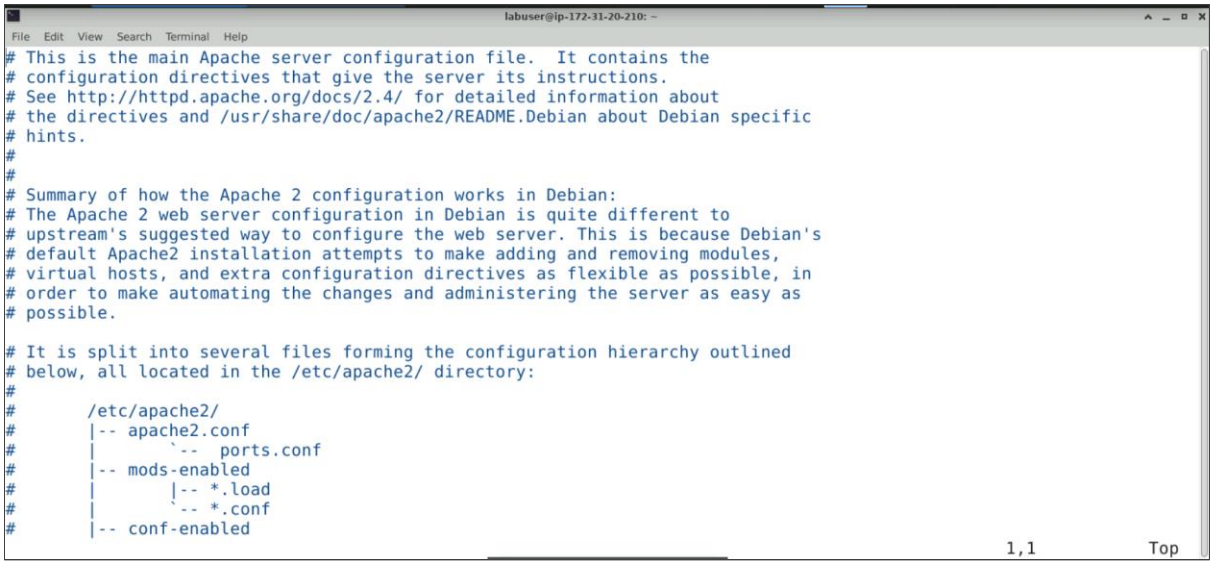
Note: If **http://localhost** shows only **Hello World!**, it indicates that Apache is serving the default index page. You can modify the index.html file in the document root (/var/www/html) to display the desired content.

1.6 Run the following command to open the configuration file **apache2.conf** in the vim editor:
sudo vim /etc/apache2/apache2.conf


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ sudo vim /etc/apache2/apache2.conf
```

The configuration file should look like this:


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
# This is the main Apache server configuration file.  It contains the
# configuration directives that give the server its instructions.
# See http://httpd.apache.org/docs/2.4/ for detailed information about
# the directives and /usr/share/doc/apache2/README.Debian about Debian specific
# hints.
#
#
# Summary of how the Apache 2 configuration works in Debian:
# The Apache 2 web server configuration in Debian is quite different to
# upstream's suggested way to configure the web server. This is because Debian's
# default Apache2 installation attempts to make adding and removing modules,
# virtual hosts, and extra configuration directives as flexible as possible, in
# order to make automating the changes and administering the server as easy as
# possible.

# It is split into several files forming the configuration hierarchy outlined
# below, all located in the /etc/apache2/ directory:
#
#       /etc/apache2/
#       |-- apache2.conf
#       |       `--  ports.conf
#       |-- mods-enabled
#       |       |-- *.load
#       |       `-- *.conf
#       |-- conf-enabled
                                                              1,1           Top
```

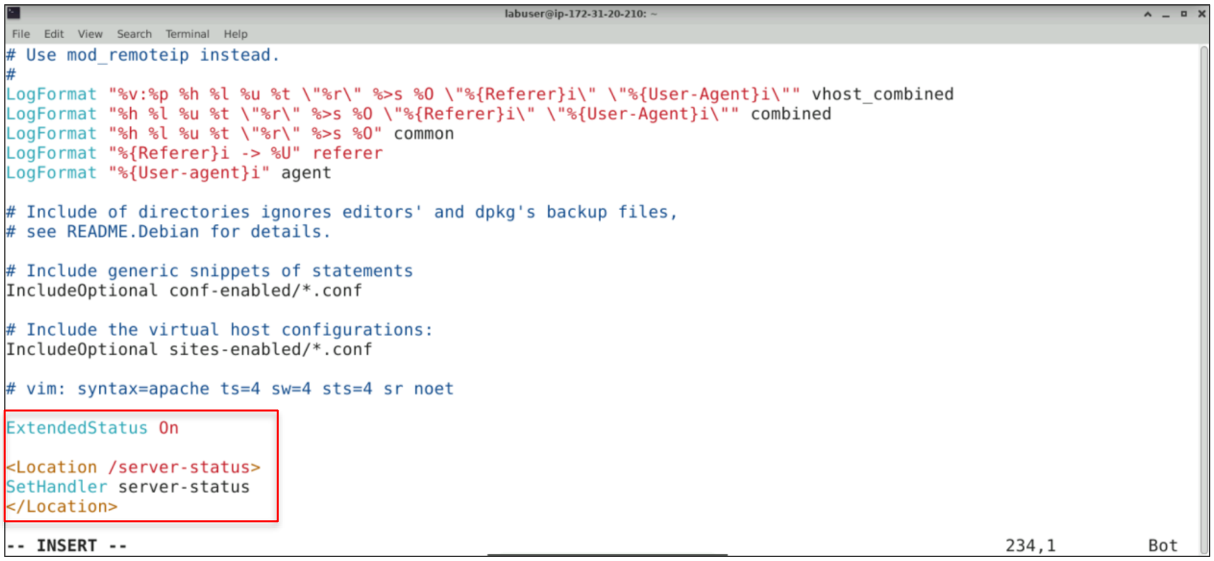
1.7 Scroll down to the end, press **I** to switch to **INSERT** mode, then copy and paste the following configuration at the end

ExtendedStatus On

<Location /server-status>

SetHandler server-status

</Location>

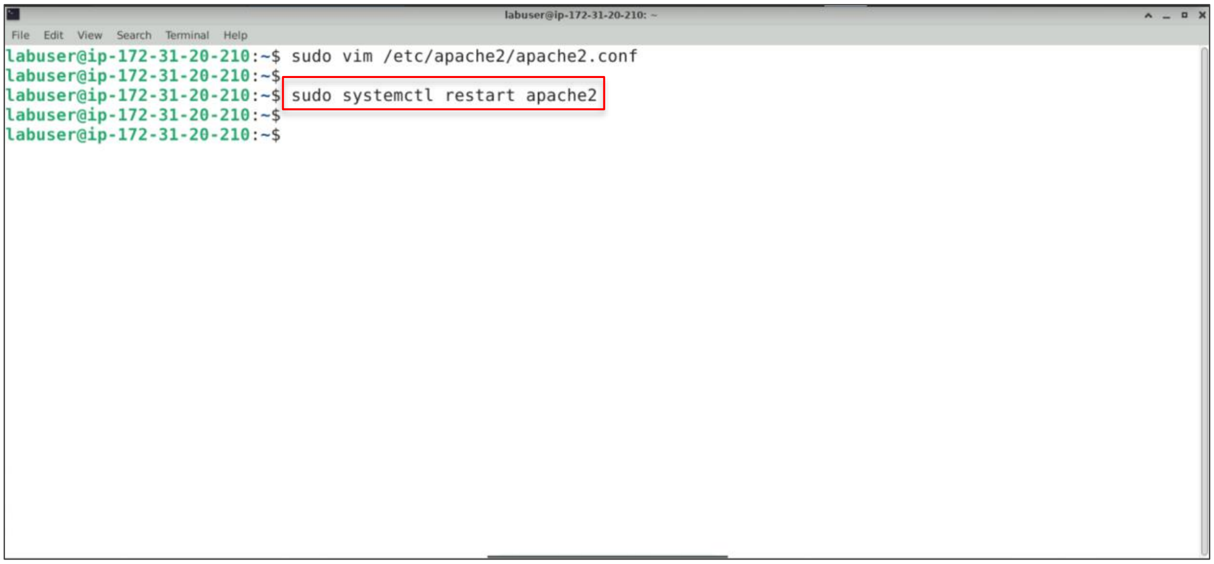

labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
# Use mod_remoteip instead.
#
LogFormat "%v:%p %h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" vhost_combined
LogFormat "%h %l %u %t \"%r\" %>s %O \"%{Referer}i\" \"%{User-Agent}i\"" combined
LogFormat "%h %l %u %t \"%r\" %>s %O" common
LogFormat "%{Referer}i -> %U" referer
LogFormat "%{User-agent}i" agent

# Include of directories ignores editors' and dpkg's backup files,
# see README.Debian for details.

# Include generic snippets of statements
IncludeOptional conf-enabled/*.conf

# Include the virtual host configurations:
IncludeOptional sites-enabled/*.conf

# vim: syntax=apache ts=4 sw=4 sts=4 sr noet

ExtendedStatus On

<Location /server-status>
SetHandler server-status
</Location>
-- INSERT --                                                                  234,1         Bot
```

Press **Esc** and type **:wq** to save and exit the file.

1.8 Execute the following command to restart Apache2:

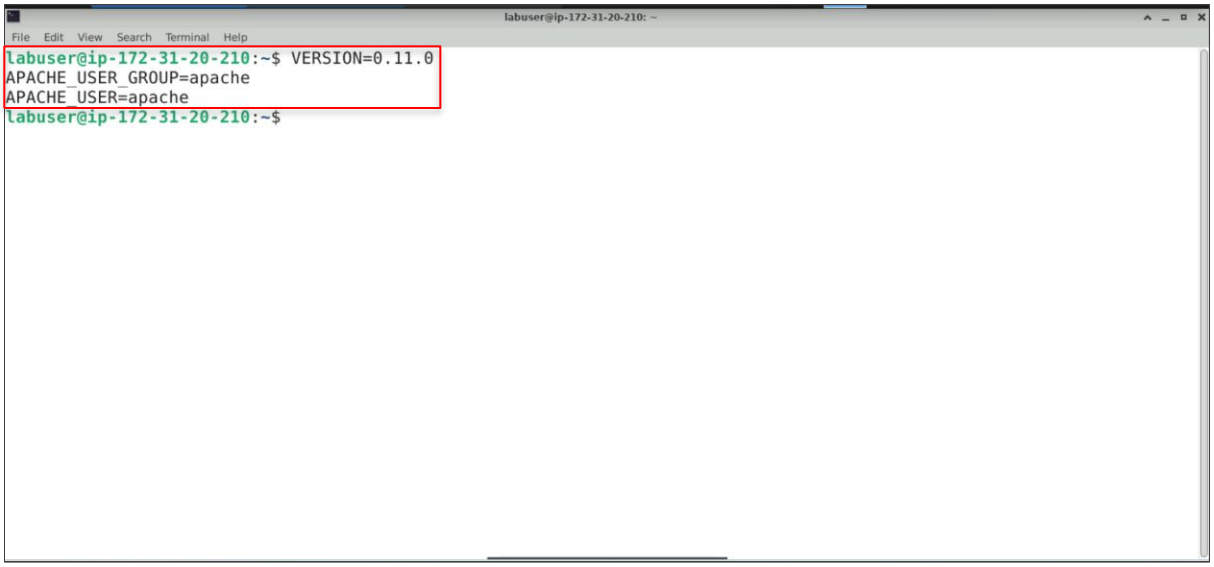
sudo systemctl restart apache2


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ sudo vim /etc/apache2/apache2.conf
labuser@ip-172-31-20-210:~$
labuser@ip-172-31-20-210:~$ sudo systemctl restart apache2
labuser@ip-172-31-20-210:~$
labuser@ip-172-31-20-210:~$
```

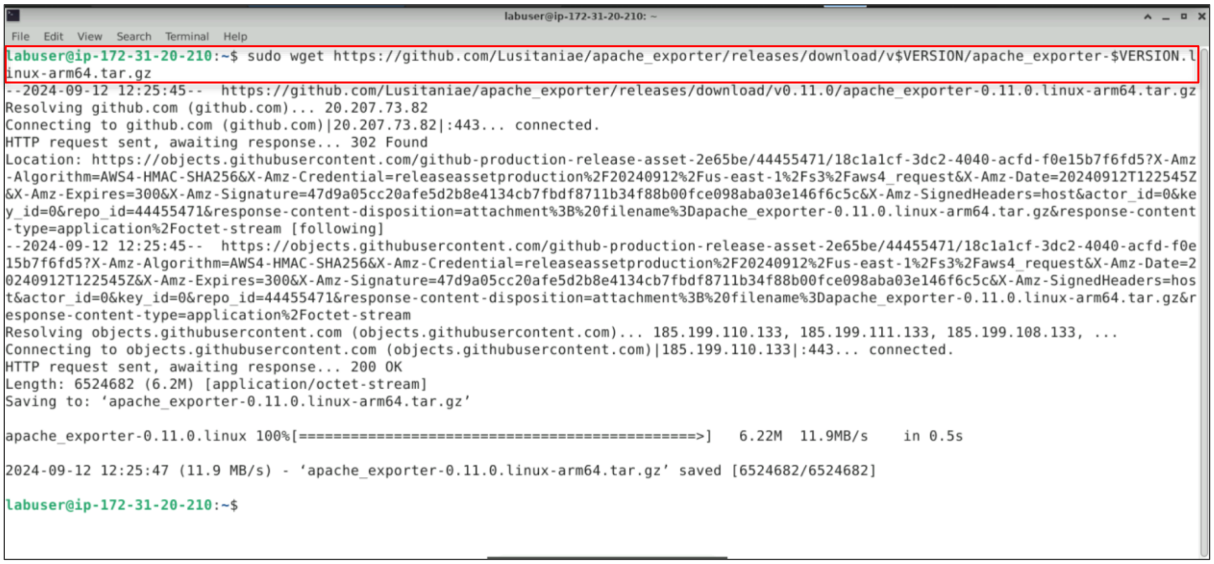
Step 2: Install and configure the Apache Exporter for Prometheus

2.1 Provide the following variables in the terminal to download the Apache Exporter:
VERSION=0.11.0
APACHE_USER_GROUP=apache
APACHE_USER=apache


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ VERSION=0.11.0
APACHE_USER_GROUP=apache
APACHE_USER=apache
labuser@ip-172-31-20-210:~$
```

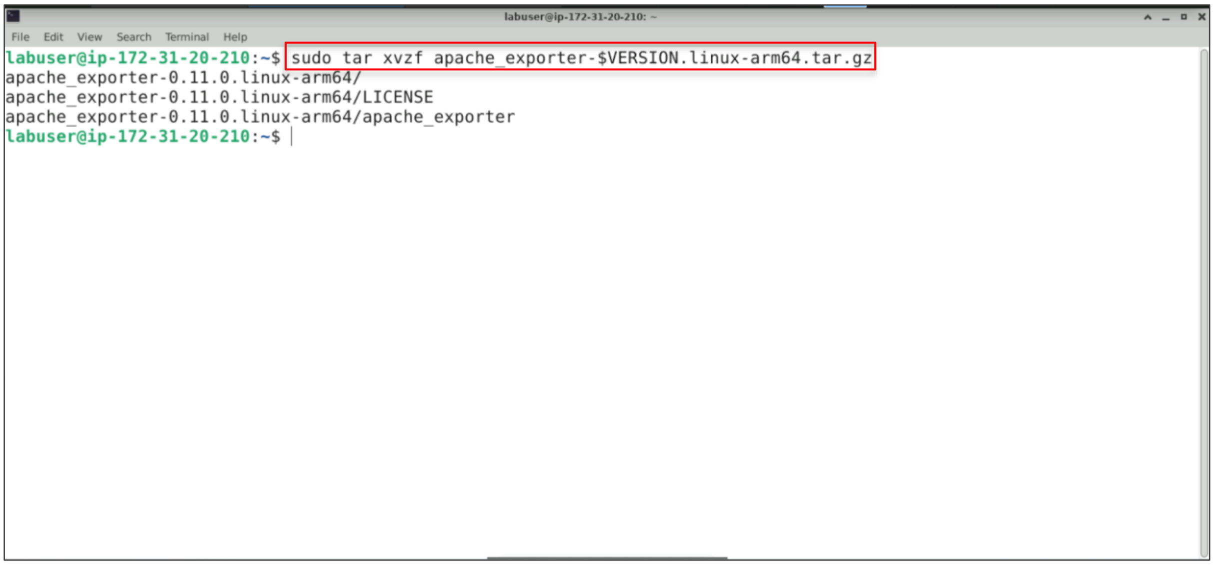
2.2 Run the following command to download the Apache Exporter file from the given URL:
sudo wget https://github.com/Lusitaniae/apache_exporter/releases/download/v$VERSION/apache_exporter-$VERSION.linux-arm64.tar.gz


labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ sudo wget https://github.com/Lusitaniae/apache_exporter/releases/download/v$VERSION/apache_exporter-$VERSION.l
inux-arm64.tar.gz
--2024-09-12 12:25:45--  https://github.com/Lusitaniae/apache_exporter/releases/download/v0.11.0/apache_exporter-0.11.0.linux-arm64.tar.gz
Resolving github.com (github.com)... 20.207.73.82
Connecting to github.com (github.com)|20.207.73.82|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://objects.githubusercontent.com/github-production-release-asset-2e65be/44455471/18c1a1cf-3dc2-4040-acfd-f0e15b7f6fd5?X-Amz
-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240912%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=20240912T122545Z
&X-Amz-Expires=300&X-Amz-Signature=47d9a05cc20afe5d2b8e4134cb7fbdf8711b34f88b00fce098aba03e146f6c5c&X-Amz-SignedHeaders=host&actor_id=0&ke
y_id=0&repo_id=44455471&response-content-disposition=attachment%3B%20filename%3Dapache_exporter-0.11.0.linux-arm64.tar.gz&response-content
-type=application%2Foctet-stream [following]
--2024-09-12 12:25:45--  https://objects.githubusercontent.com/github-production-release-asset-2e65be/44455471/18c1a1cf-3dc2-4040-acfd-f0e
15b7f6fd5?X-Amz-Algorithm=AWS4-HMAC-SHA256&X-Amz-Credential=releaseassetproduction%2F20240912%2Fus-east-1%2Fs3%2Faws4_request&X-Amz-Date=2
0240912T122545Z&X-Amz-Expires=300&X-Amz-Signature=47d9a05cc20afe5d2b8e4134cb7fbdf8711b34f88b00fce098aba03e146f6c5c&X-Amz-SignedHeaders=hos
t&actor_id=0&key_id=0&repo_id=44455471&response-content-disposition=attachment%3B%20filename%3Dapache_exporter-0.11.0.linux-arm64.tar.gz&r
esponse-content-type=application%2Foctet-stream
Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.110.133, 185.199.111.133, 185.199.108.133, ...
Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.110.133|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6524682 (6.2M) [application/octet-stream]
Saving to: ‘apache_exporter-0.11.0.linux-arm64.tar.gz’

apache_exporter-0.11.0.linux 100%[=============================================>]   6.22M  11.9MB/s    in 0.5s

2024-09-12 12:25:47 (11.9 MB/s) - ‘apache_exporter-0.11.0.linux-arm64.tar.gz’ saved [6524682/6524682]

labuser@ip-172-31-20-210:~$
```

2.3 Run the command given below to extract the downloaded Apache Exporter:

sudo tar xvzf apache_exporter-$VERSION.linux-arm64.tar.gz



labuser@ip-172-31-20-210: ~

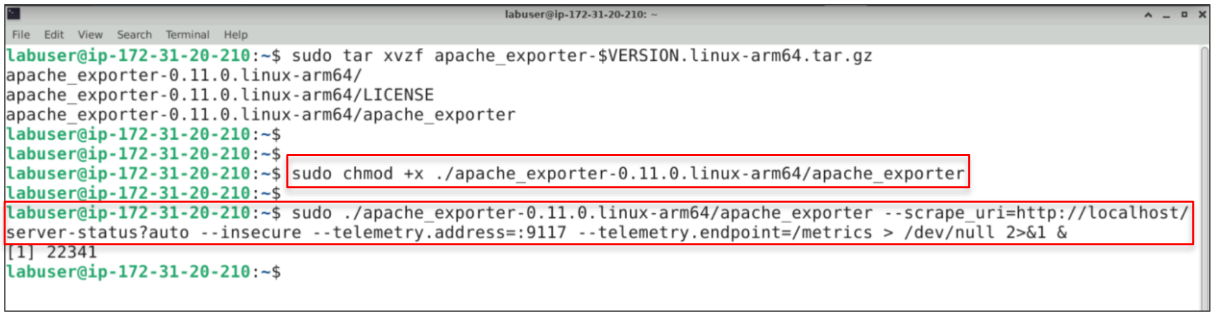
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ sudo tar xvzf apache_exporter-$VERSION.linux-arm64.tar.gz
apache_exporter-0.11.0.linux-arm64/
apache_exporter-0.11.0.linux-arm64/LICENSE
apache_exporter-0.11.0.linux-arm64/apache_exporter
labuser@ip-172-31-20-210:~$ 
```

2.4 Run the following commands to change the file permissions of Apache Exporter and make it executable, then start it in the background:

sudo chmod +x ./apache_exporter-0.11.0.linux-arm64/apache_exporter

sudo ./apache_exporter-0.11.0.linux-arm64/apache_exporter --scrape_uri=http://localhost/server-status?auto --insecure --telemetry.address=:9117 --telemetry.endpoint=/metrics > /dev/null 2>&1 &

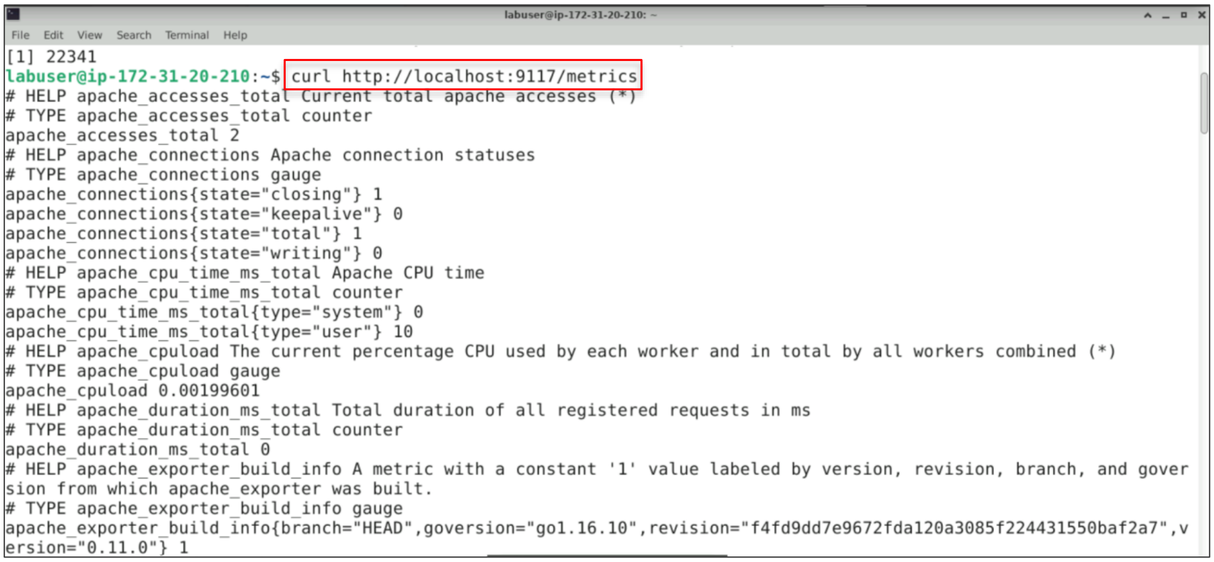


labuser@ip-172-31-20-210: ~

File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ sudo tar xvzf apache_exporter-$VERSION.linux-arm64.tar.gz
apache_exporter-0.11.0.linux-arm64/
apache_exporter-0.11.0.linux-arm64/LICENSE
apache_exporter-0.11.0.linux-arm64/apache_exporter
labuser@ip-172-31-20-210:~$
labuser@ip-172-31-20-210:~$
labuser@ip-172-31-20-210:~$ sudo chmod +x ./apache_exporter-0.11.0.linux-arm64/apache_exporter
labuser@ip-172-31-20-210:~$
labuser@ip-172-31-20-210:~$ sudo ./apache_exporter-0.11.0.linux-arm64/apache_exporter --scrape_uri=http://localhost/
server-status?auto --insecure --telemetry.address=:9117 --telemetry.endpoint=/metrics > /dev/null 2>&1 &
[1] 22341
labuser@ip-172-31-20-210:~$
```

2.5 Run the following command to verify whether metrics are being fetched from the Apache Exporter running on the local machine:
curl http://localhost:9117/metrics

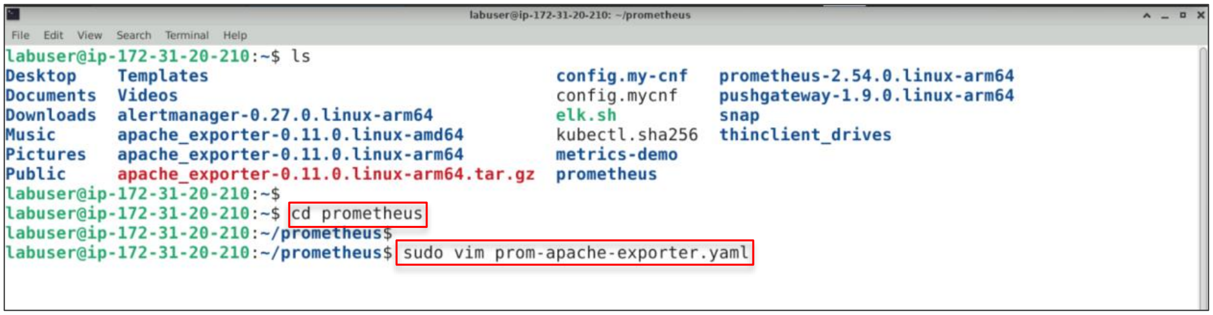

labuser@ip-172-31-20-210: ~
File Edit View Search Terminal Help

```
[1] 22341
labuser@ip-172-31-20-210:~$ curl http://localhost:9117/metrics
# HELP apache_accesses_total Current total apache accesses (*)
# TYPE apache_accesses_total counter
apache_accesses_total 2
# HELP apache_connections Apache connection statuses
# TYPE apache_connections gauge
apache_connections{state="closing"} 1
apache_connections{state="keepalive"} 0
apache_connections{state="total"} 1
apache_connections{state="writing"} 0
# HELP apache_cpu_time_ms_total Apache CPU time
# TYPE apache_cpu_time_ms_total counter
apache_cpu_time_ms_total{type="system"} 0
apache_cpu_time_ms_total{type="user"} 10
# HELP apache_cpuload The current percentage CPU used by each worker and in total by all workers combined (*)
# TYPE apache_cpuload gauge
apache_cpuload 0.00199601
# HELP apache_duration_ms_total Total duration of all registered requests in ms
# TYPE apache_duration_ms_total counter
apache_duration_ms_total 0
# HELP apache_exporter_build_info A metric with a constant '1' value labeled by version, revision, branch, and gover
sion from which apache_exporter was built.
# TYPE apache_exporter_build_info gauge
apache_exporter_build_info{branch="HEAD",goversion="go1.16.10",revision="f4fd9dd7e9672fda120a3085f224431550baf2a7",v
ersion="0.11.0"} 1
```

Note: You can also access these metrics in the browser using the same URL without the command **curl**.

Step 3: Configure Prometheus to scrape metrics from the Apache Exporter

3.1 Run the following command to change the directory to **prometheus**, then create and open the file **prom-apache-exporter.yaml** using the **Vim** editor:
cd prometheus
sudo vim prom-apache-exporter.yaml

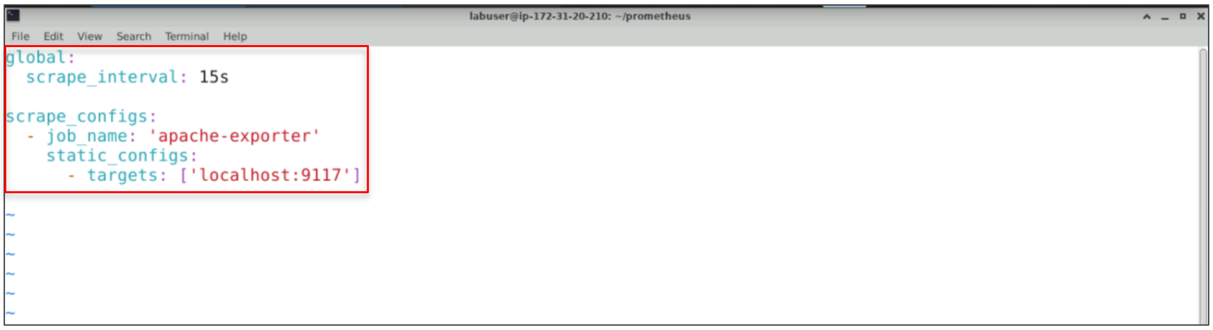

labuser@ip-172-31-20-210: ~/prometheus
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~$ ls
Desktop    Templates                                config.my-cnf   prometheus-2.54.0.linux-arm64
Documents  Videos                                   config.mycnf    pushgateway-1.9.0.linux-arm64
Downloads  alertmanager-0.27.0.linux-arm64          elk.sh          snap
Music      apache_exporter-0.11.0.linux-amd64       kubectl.sha256  thinclient_drives
Pictures   apache_exporter-0.11.0.linux-arm64       metrics-demo
Public     apache_exporter-0.11.0.linux-arm64.tar.gz  prometheus
labuser@ip-172-31-20-210:~$
labuser@ip-172-31-20-210:~$ cd prometheus
labuser@ip-172-31-20-210:~/prometheus$
labuser@ip-172-31-20-210:~/prometheus$ sudo vim prom-apache-exporter.yaml
```

Note: Ensure that Prometheus is already installed

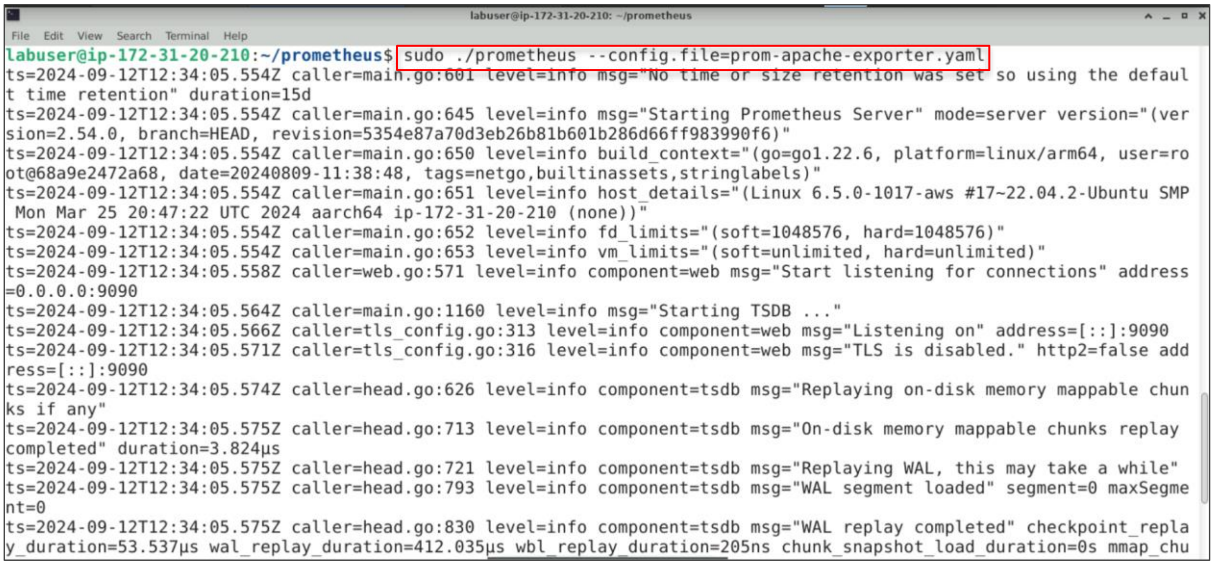
3.2 Switch to **INSERT** mode, copy and paste the following configuration, then save and exit the file:

```
global:
  scrape_interval: 15s

scrape_configs:
  - job_name: 'apache-exporter'
    static_configs:
      - targets: ['localhost:9117']
```


labuser@ip-172-31-20-210: ~/prometheus
File Edit View Search Terminal Help

```
global:
  scrape_interval: 15s

scrape_configs:
  - job_name: 'apache-exporter'
    static_configs:
      - targets: ['localhost:9117']

~
~
~
~
~
~
```

3.3 Run the following command to explore the Prometheus UI:
sudo ./prometheus --config.file=prom-apache-exporter.yaml

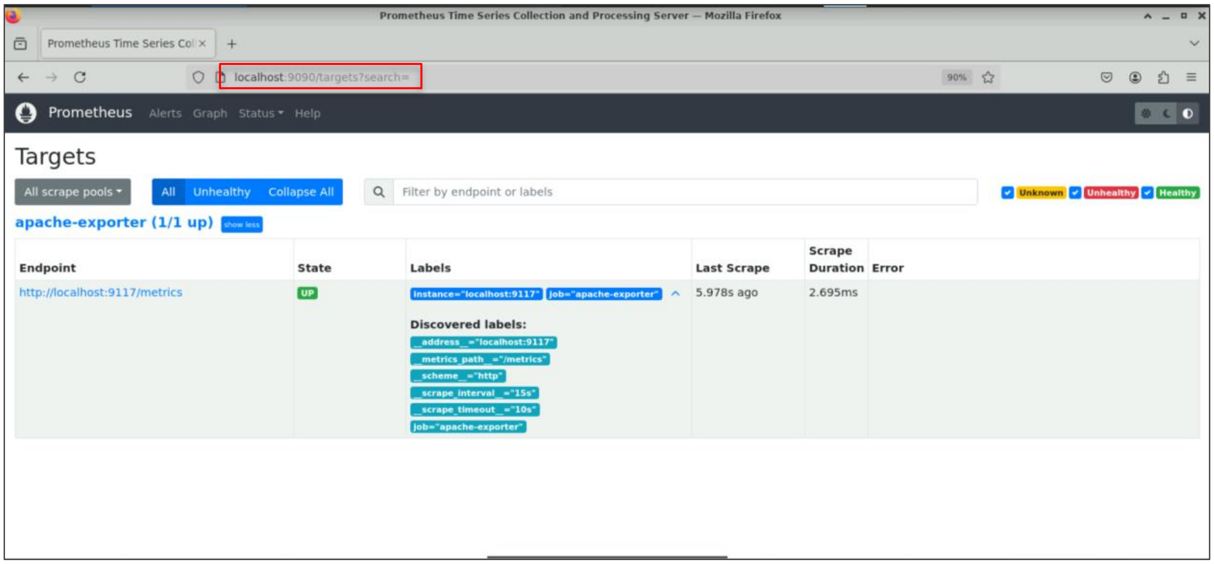

labuser@ip-172-31-20-210: ~/prometheus
File Edit View Search Terminal Help

```
labuser@ip-172-31-20-210:~/prometheus$ sudo ./prometheus --config.file=prom-apache-exporter.yaml
ts=2024-09-12T12:34:05.554Z caller=main.go:601 level=info msg="No time or size retention was set so using the defaul
t time retention" duration=15d
ts=2024-09-12T12:34:05.554Z caller=main.go:645 level=info msg="Starting Prometheus Server" mode=server version="(ver
sion=2.54.0, branch=HEAD, revision=5354e87a70d3eb26b81b601b286d66ff983990f6)"
ts=2024-09-12T12:34:05.554Z caller=main.go:650 level=info build_context="(go=go1.22.6, platform=linux/arm64, user=ro
ot@68a9e2472a68, date=20240809-11:38:48, tags=netgo,builtinassets,stringlabels)"
ts=2024-09-12T12:34:05.554Z 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 aarch64 ip-172-31-20-210 (none))"
ts=2024-09-12T12:34:05.554Z caller=main.go:652 level=info fd_limits="(soft=1048576, hard=1048576)"
ts=2024-09-12T12:34:05.554Z caller=main.go:653 level=info vm_limits="(soft=unlimited, hard=unlimited)"
ts=2024-09-12T12:34:05.558Z caller=web.go:571 level=info component=web msg="Start listening for connections" address
=0.0.0.0:9090
ts=2024-09-12T12:34:05.564Z caller=main.go:1160 level=info msg="Starting TSDB ..."
ts=2024-09-12T12:34:05.566Z caller=tls_config.go:313 level=info component=web msg="Listening on" address=[::]:9090
ts=2024-09-12T12:34:05.571Z caller=tls_config.go:316 level=info component=web msg="TLS is disabled." http2=false add
ress=[::]:9090
ts=2024-09-12T12:34:05.574Z caller=head.go:626 level=info component=tsdb msg="Replaying on-disk memory mappable chun
ks if any"
ts=2024-09-12T12:34:05.575Z caller=head.go:713 level=info component=tsdb msg="On-disk memory mappable chunks replay
completed" duration=3.824µs
ts=2024-09-12T12:34:05.575Z caller=head.go:721 level=info component=tsdb msg="Replaying WAL, this may take a while"
ts=2024-09-12T12:34:05.575Z caller=head.go:793 level=info component=tsdb msg="WAL segment loaded" segment=0 maxSegme
nt=0
ts=2024-09-12T12:34:05.575Z caller=head.go:830 level=info component=tsdb msg="WAL replay completed" checkpoint_repla
y_duration=53.537µs wal_replay_duration=412.035µs wbl_replay_duration=205ns chunk_snapshot_load_duration=0s mmap_chu
```

Note: You can also run Prometheus in the background using the command **sudo ./prometheus --config.file=prom-apache-exporter.yaml > /dev/null 2>&1 &** However, this command will not detect port conflict issues.

Step 4: Run PromQL queries in the Prometheus UI to analyze Apache web server metrics

4.1 Open the preferred browser and navigate to the URL **http://localhost:9090/targets** to explore the Apache Exporter on the Prometheus UI

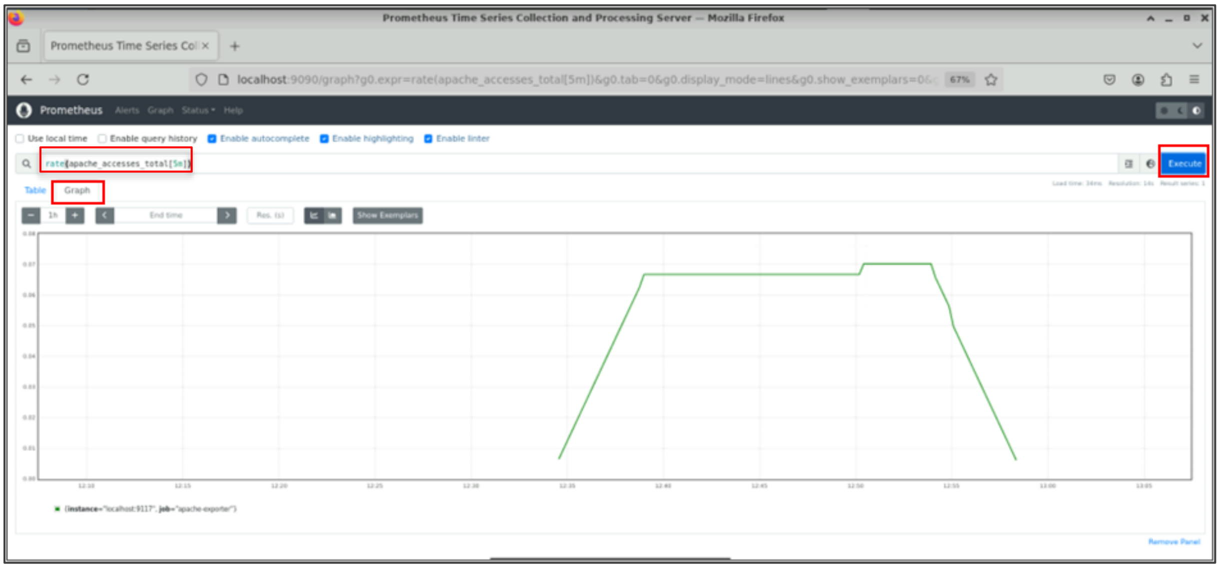

Prometheus Time Series Collection and Processing Server — Mozilla Firefox
Prometheus Time Series Col ×
localhost:9090/targets?search=
90%
Prometheus
Alerts
Graph
Status
Help
All scrape pools
All
Unhealthy
Collapse All
Filter by endpoint or labels
Unknown
Unhealthy
Healthy
show less

Targets

apache-exporter (1/1 up)

Endpoint	State	Labels	Last Scrape	Scrape Duration	Error
http://localhost:9117/metrics	UP	instance="localhost:9117" job="apache-exporter" Discovered labels: __address__="localhost:9117" __metrics_path__="/metrics" __scheme__="http" __scrape_interval__="15s" __scrape_timeout__="10s" job="apache-exporter"	5.978s ago	2.695ms	

4.2 Navigate to the **Graph** section in Prometheus, enter the following query in the expression bar, and click the **Execute** button to calculate the rate of Apache requests per second over the last 5 minutes:
rate(apache_accesses_total[5m])

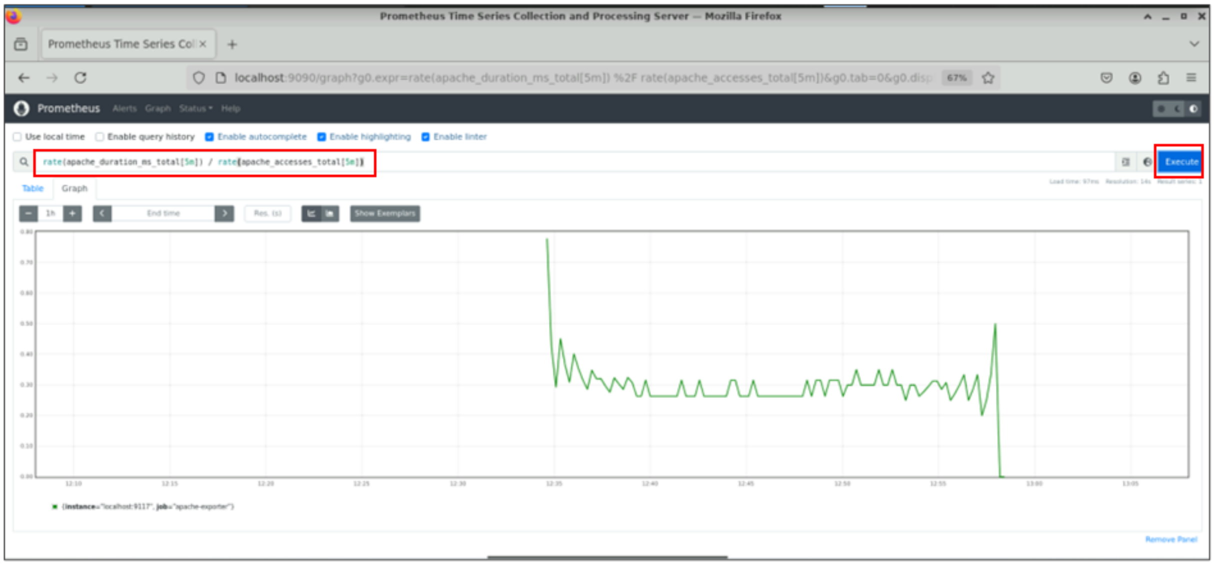

Prometheus Time Series Collection and Processing Server — Mozilla Firefox
Prometheus Time Series Col ×
localhost:9090/graph?g0.expr=rate(apache_accesses_total[5m])&g0.tab=0&g0.display_mode=lines&g0.show_exemplars=0&
67%
Prometheus
Alerts
Graph
Status
Help
Execute
Table
Graph
1h
End time
Res. (s)
Show Exemplars
0.08
0.07
0.06
0.05
0.04
0.03
0.02
0.01
0.00
12:10
12:15
12:20
12:25
12:30
12:35
12:40
12:45
12:50
12:55
13:00
13:05
Remove Panel

☐ Use local time ☐ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

```
rate(apache_accesses_total[5m])
```

{instance="localhost:9117", job="apache-exporter"}

4.3 Enter the following query in the expression bar and click on the **Execute** button to measure the average response time of Apache requests in milliseconds over the last 5 minutes:
rate(apache_duration_ms_total[5m]) / rate(apache_accesses_total[5m])

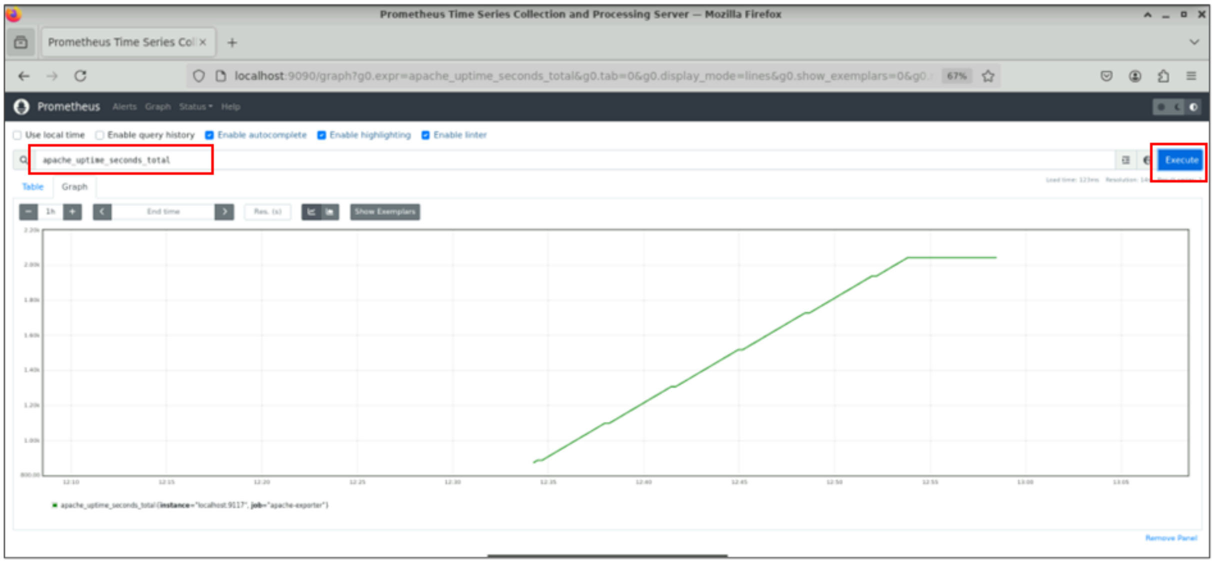

Prometheus Time Series Collection and Processing Server — Mozilla Firefox
Prometheus Time Series Col ×
localhost:9090/graph?g0.expr=rate(apache_duration_ms_total[5m]) %2F rate(apache_accesses_total[5m])&g0.tab=0&g0.disp
67%
Prometheus
Alerts
Graph
Status
Help
Table
Graph
1h
End time
Res. (s)
Show Exemplars
Execute
Remove Panel

□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

```
rate(apache_duration_ms_total[5m]) / rate(apache_accesses_total[5m])
```

0.80
0.70
0.60
0.50
0.40
0.30
0.20
0.10
0.00
12:10
12:15
12:20
12:25
12:30
12:35
12:40
12:45
12:50
12:55
13:00
13:05
{instance="localhost:9117", job="apache-exporter"}

4.4 Enter the following query in the expression bar and click the **Execute** button to display the total uptime of the Apache server in seconds:
apache_uptime_seconds_total

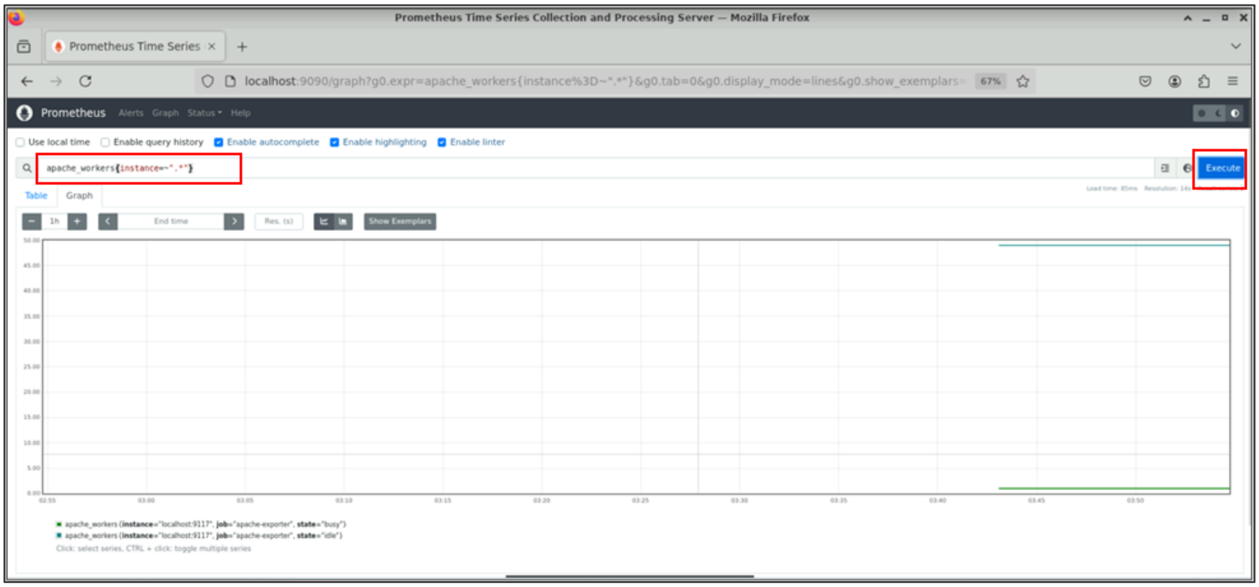

Prometheus Time Series Collection and Processing Server — Mozilla Firefox
Prometheus Time Series Col ×
localhost:9090/graph?g0.expr=apache_uptime_seconds_total&g0.tab=0&g0.display_mode=lines&g0.show_exemplars=0&g0.
67%
Prometheus
Alerts
Graph
Status
Help
Table
Graph
1h
End time
Res. (s)
Show Exemplars
Execute
Remove Panel

□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

```
apache_uptime_seconds_total
```

2.20k
2.00k
1.80k
1.60k
1.40k
1.20k
1.00k
800.00
12:10
12:15
12:20
12:25
12:30
12:35
12:40
12:45
12:50
12:55
13:00
13:05
apache_uptime_seconds_total{instance="localhost:9117", job="apache-exporter"}

4.5 Enter the following query in the expression bar and click the **Execute** button to display the count of Apache workers for each instance:
apache_workers{instance=~".*"}



Prometheus Time Series Collection and Processing Server — Mozilla Firefox

localhost:9090/graph?g0.expr=apache_workers{instance%3D~".*"}&g0.tab=0&g0.display_mode=lines&g0.show_exemplars=

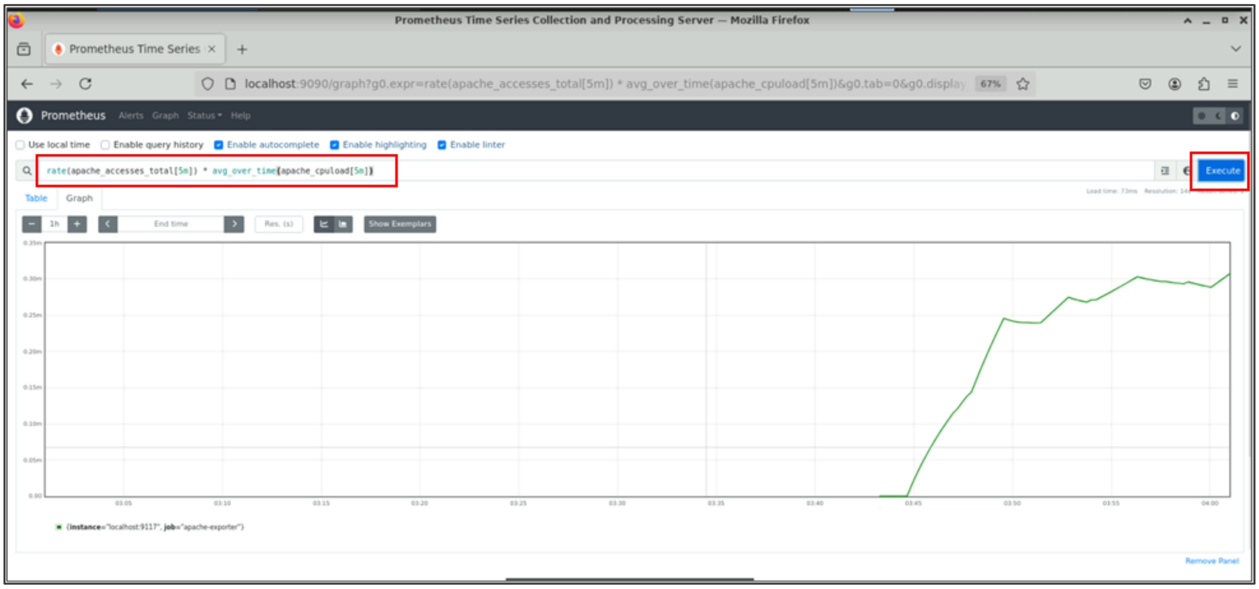
□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

```
apache_workers{instance=~".*"}
```

🟩 apache_workers{**instance**="localhost:9117", **job**="apache-exporter", **state**="busy"}
🟦 apache_workers{**instance**="localhost:9117", **job**="apache-exporter", **state**="idle"}

Click: select series, CTRL + click: toggle multiple series

4.6 Enter the following query in the expression bar and click the **Execute** button to monitor Apache's request rate and CPU load over the same time interval:
rate(apache_accesses_total[5m]) * avg_over_time(apache_cpuload[5m])



Prometheus Time Series Collection and Processing Server — Mozilla Firefox

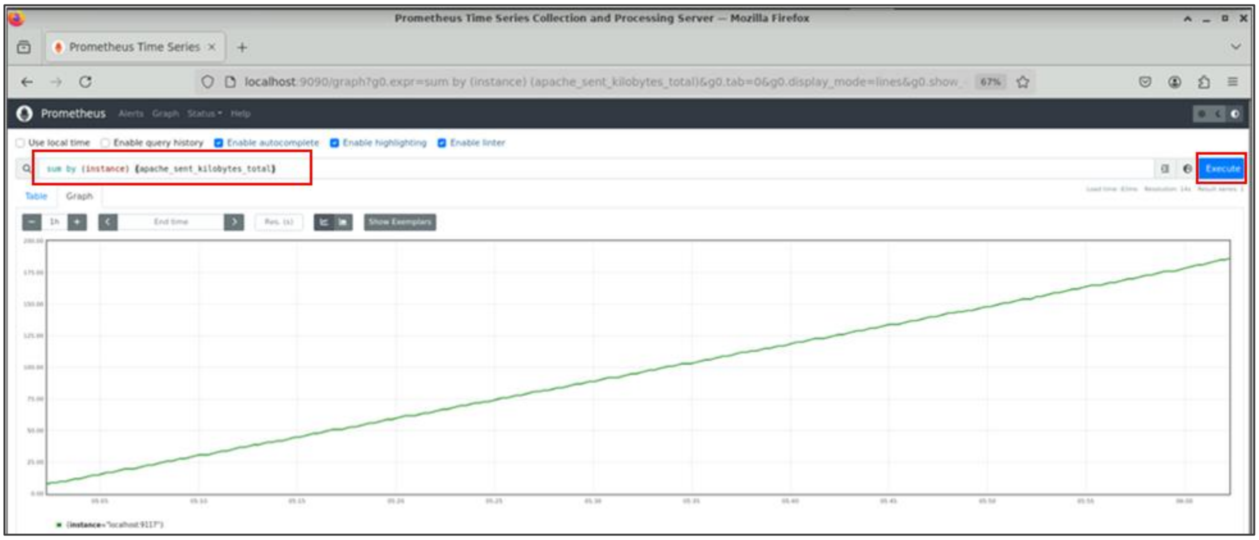
localhost:9090/graph?g0.expr=rate(apache_accesses_total[5m]) * avg_over_time(apache_cpuload[5m])&g0.tab=0&g0.display

□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

```
rate(apache_accesses_total[5m]) * avg_over_time(apache_cpuload[5m])
```

🟩 {**instance**="localhost:9117", **job**="apache-exporter"}

4.7 Enter the following query in the expression bar and click the **Execute** button to display the total amount of data sent (in kilobytes) by the Apache server for each instance: **sum by (instance) (apache_sent_kilobytes_total)**



Prometheus Time Series Collection and Processing Server — Mozilla Firefox

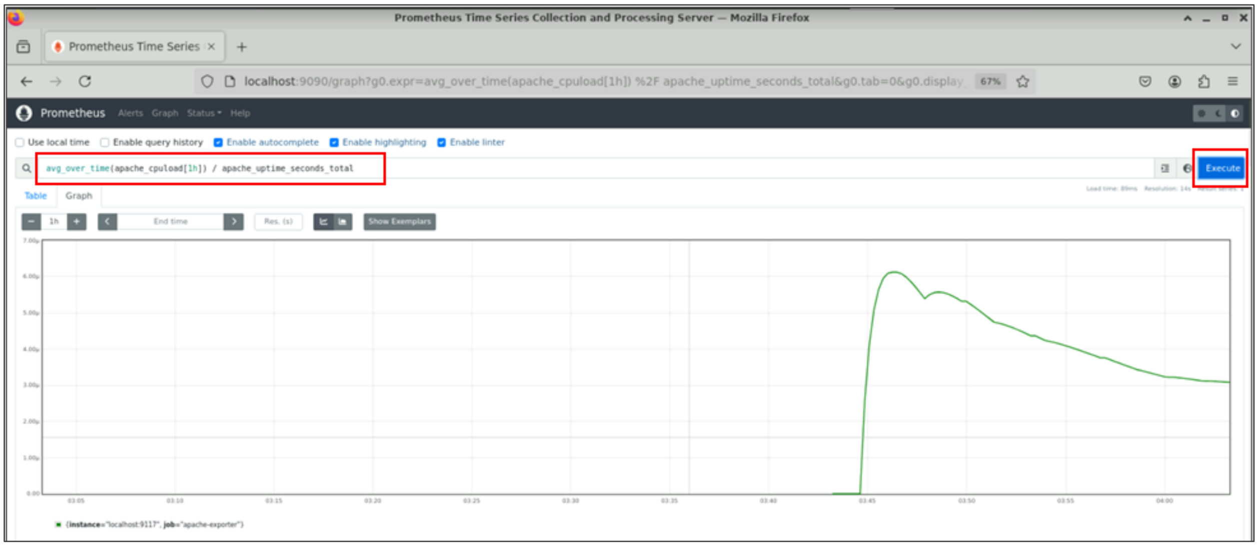
localhost:9090/graph?g0.expr=sum by (instance) (apache_sent_kilobytes_total)&g0.tab=0&g0.display_mode=lines&g0.show_

□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

```
sum by (instance) (apache_sent_kilobytes_total)
```

■ {instance="localhost:9117"}

4.8 Enter the following query in the expression bar and click the **Execute** button to monitor the average CPU load of Apache over the last hour relative to its total uptime: **avg_over_time(apache_cpuload[1h]) / apache_uptime_seconds_total**



Prometheus Time Series Collection and Processing Server — Mozilla Firefox

localhost:9090/graph?g0.expr=avg_over_time(apache_cpuload[1h]) %2F apache_uptime_seconds_total&g0.tab=0&g0.display_

□ Use local time □ Enable query history ☑ Enable autocomplete ☑ Enable highlighting ☑ Enable linter

```
avg_over_time(apache_cpuload[1h]) / apache_uptime_seconds_total
```

■ {instance="localhost:9117", job="apache-exporter"}

By following these steps, you have successfully set up and configured an Apache web server along with the Prometheus monitoring system, including the Apache Exporter setup to collect and visualize Apache web server metrics. These metrics are accessible through a web interface for real-time monitoring and performance analysis using custom PromQL queries.