**Project-3**

**Integrate Grafana with Linux Server for high cpu utilization and create a graph in Grafana.**

**Table of content**

* Introduction
* What is Grafana ?
* Architecture overview
* Step-by-step guide

**Introduction**

In today’s IT infrastructure, **monitoring server performance** is critical to ensure the availability, stability, and efficiency of applications and services. One of the key performance metrics in any Linux server environment is **CPU utilization**. High CPU usage can lead to server slowdowns, service outages, and poor user experience. Therefore, continuous monitoring and visualization of CPU utilization is essential for **system administrators** and **DevOps engineers**.

This project focuses on **integrating Grafana with a Linux server** to monitor **high CPU utilization** and visualize the data using **interactive dashboards and graphs**.

**What is Grafana ?**

**Grafana** is an open-source analytics and interactive visualization web application. It allows you to **query**, **visualize**, **alert on**, and **explore** your metrics, logs, and traces from multiple data sources, all in one place.

It is widely used for **monitoring infrastructure**, **server performance**, **application monitoring**, **IoT device monitoring**, and even **business intelligence dashboards.**

**Key features of Grafana**

| **Feature** | **Description** |
| --- | --- |
| **Data Visualization** | Grafana converts raw data into interactive charts, graphs, and alerts. |
| **Supports Multiple Data Sources** | You can connect Grafana with databases like Prometheus, InfluxDB, MySQL, PostgreSQL, AWS CloudWatch, Elasticsearch, etc. |
| **Custom Dashboards** | Create fully customizable and dynamic dashboards to visualize data from different sources. |
| **Alerting** | Set conditions on your data and get alerts (Email, Slack, Webhooks, etc) when something goes wrong (like CPU > 90%). |
| **Templating** | Allows you to create dashboards with variables so that you can easily switch between servers, instances, or regions. |
| **User Management** | Manage user roles (Admin, Editor, Viewer) with access control and sharing features. |
| **Plugins** | Grafana supports plugins for different visualization types, data sources, and apps. |

**How Grafana works**

 **Data Collection:**  
Grafana does **not store data** itself (except dashboard settings). It connects to external **data sources** like:

* Time series databases (Prometheus, InfluxDB)
* SQL databases (MySQL, PostgreSQL)
* Cloud services (AWS CloudWatch, Google Cloud Monitoring)
* Logging systems (Loki, Elasticsearch)

 **Query Engine:**  
Grafana uses query languages (specific to each data source) to pull data in real-time.

 **Visualization:**  
Grafana processes the data and displays it as **graphs**, **charts**, **tables**, **gauges**, etc.

 **Dashboard:**  
You can combine multiple visualizations into one or more **dashboards** for monitoring at a glance.

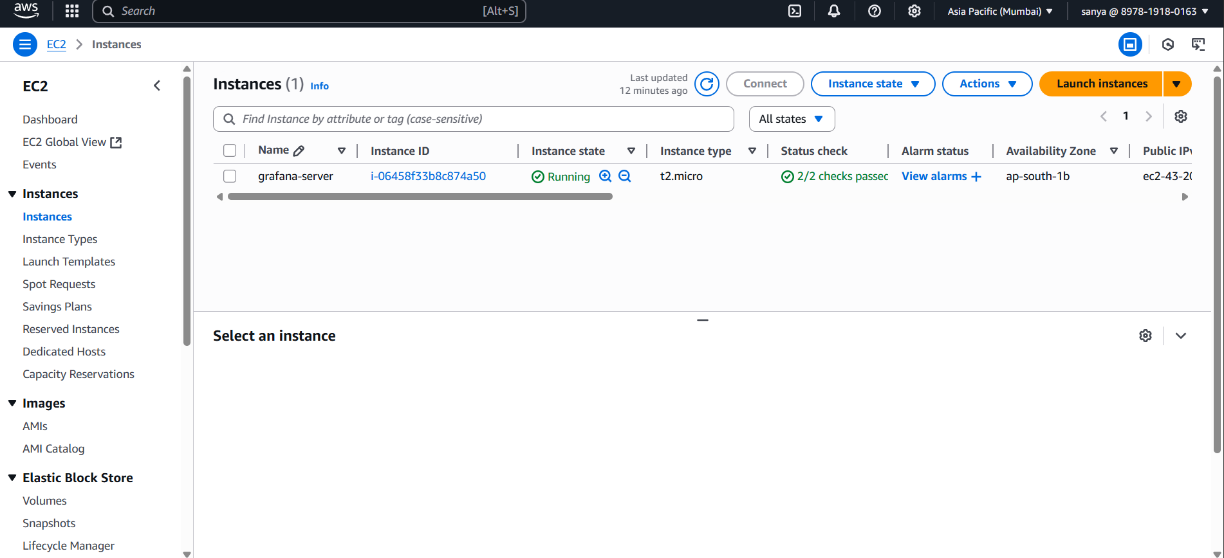
 **Alerts:**  
You can set **thresholds** on your data to trigger alerts when certain conditions are met

**Architecture overview**

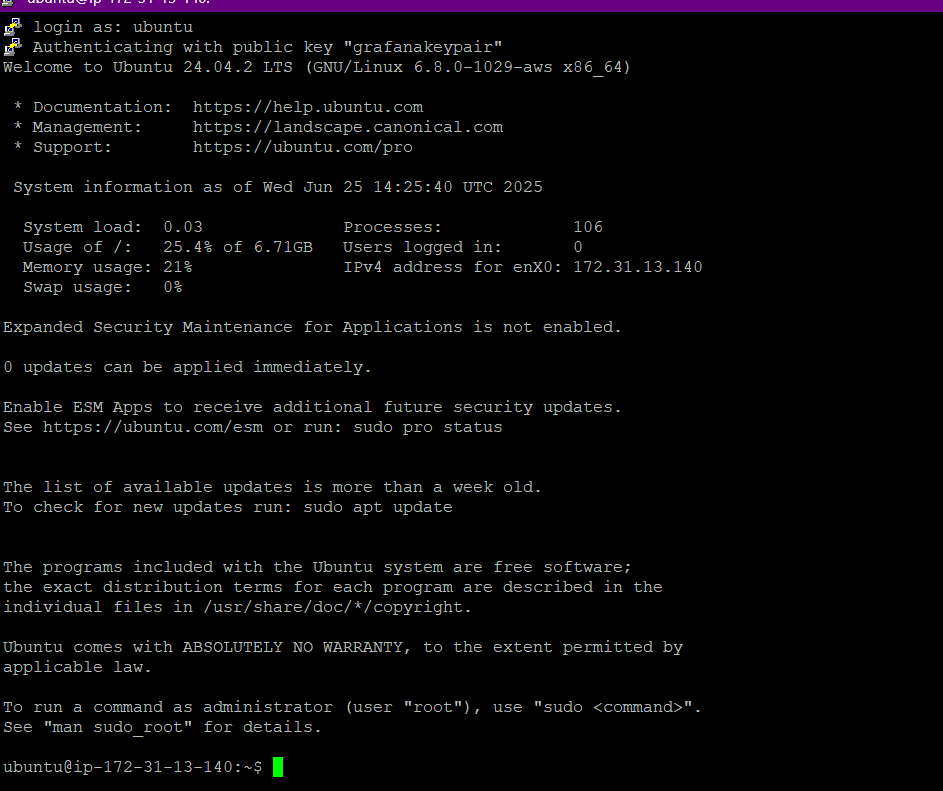
****

**Step – by – step guide**

* Login in AWS account and go to the ec2.
* In ec2 launch the instance in Mumbai – ap south1 region.
* Create the new key
* Now launch the instance.



* Now instance is launched connect it with putty.



* Your screen will look like this.
* Now run the given commands to install the Grafana.

1. sudo apt update && sudo apt upgrade -y

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

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

4.sudo apt-get install -y apt-transport-https

sudo apt-get install -y software-properties-common wget

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

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

sudo apt-get update

sudo apt-get install Grafana

5. sudo systemctl daemon-reexec

sudo systemctl start grafana-server

sudo systemctl enable grafana-server

6. ls /lib/systemd/system/grafana-server.service

7. sudo apt remove grafana

sudo apt update

sudo apt install grafana

8. sudo apt update && sudo apt upgrade -y

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

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

11. sudo apt update

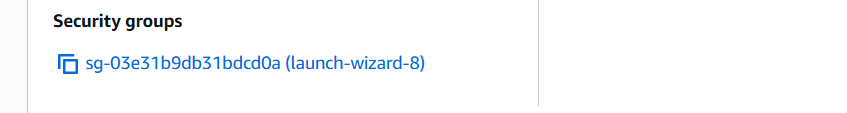
12. sudo apt install grafana -y

13. sudo systemctl start grafana-server

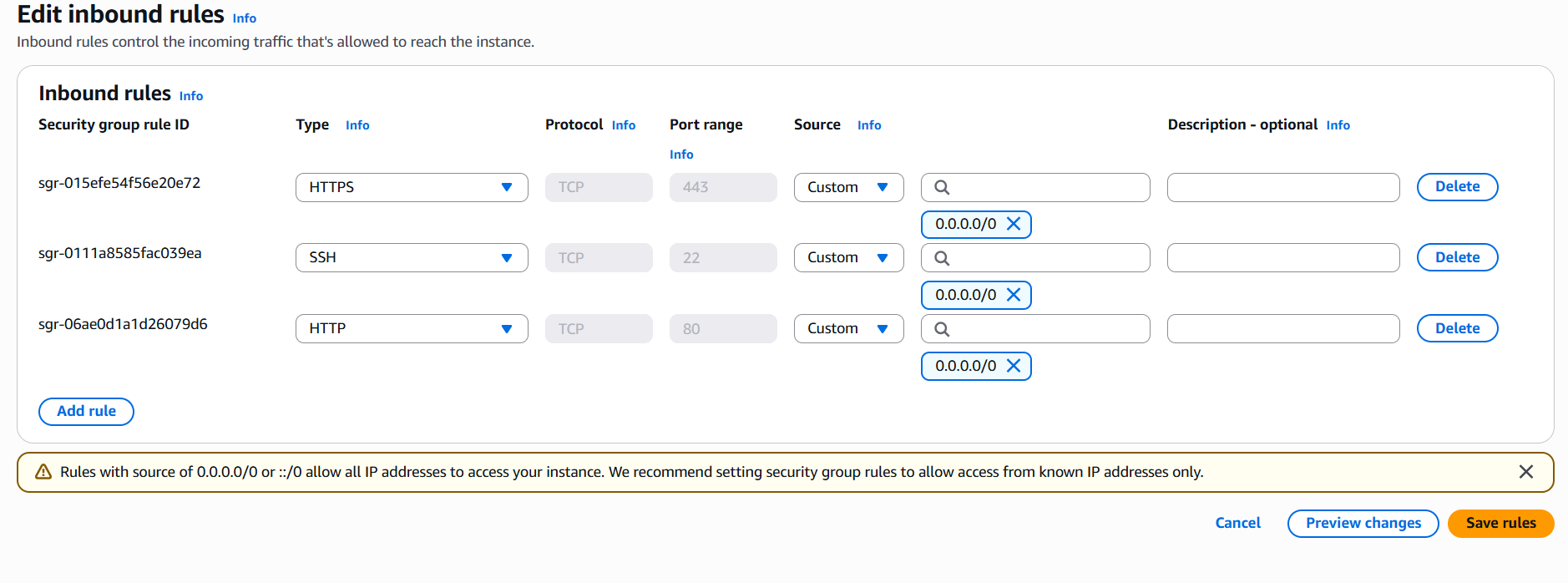
14. sudo systemctl enable grafana-server

15. sudo systemctl status grafana-server

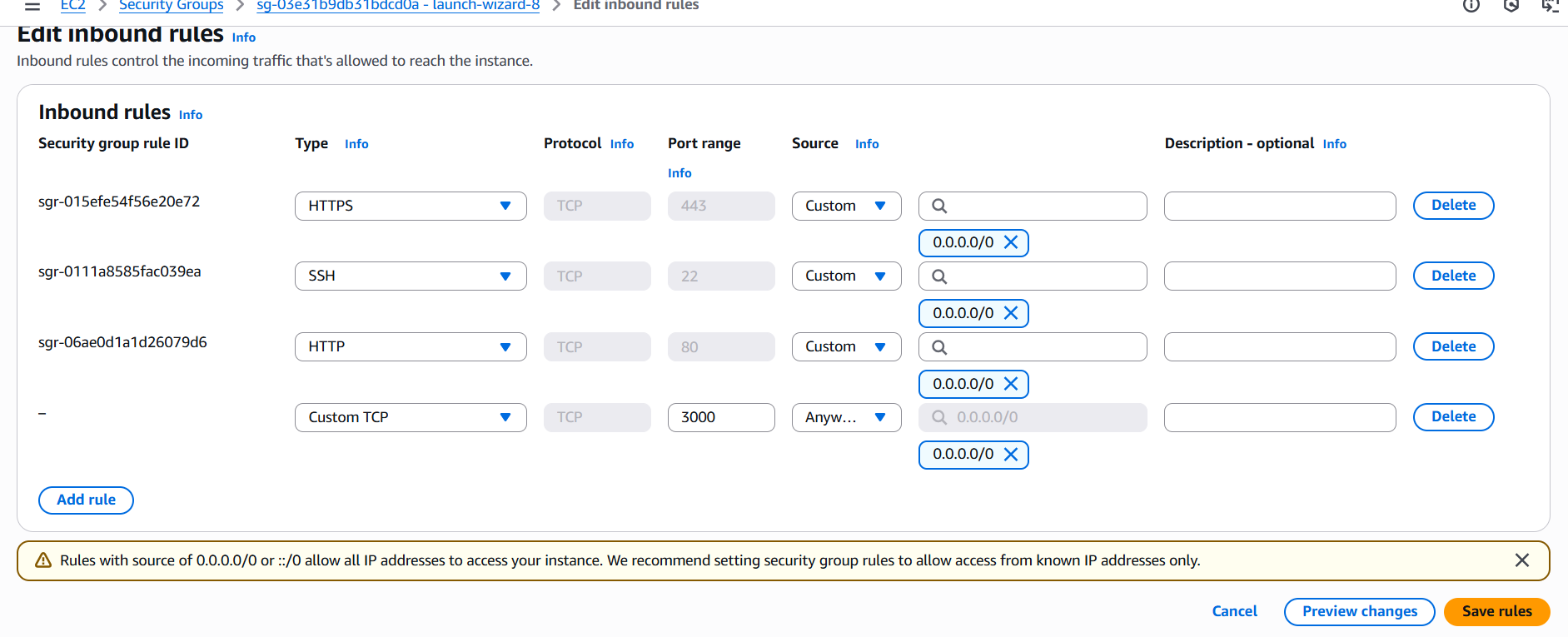
* Now run the code in the given linux server.
* After running all the commands go to the instance which you have created earlier and go to the security groups.



* Click on the security group and edit the inbound rules.



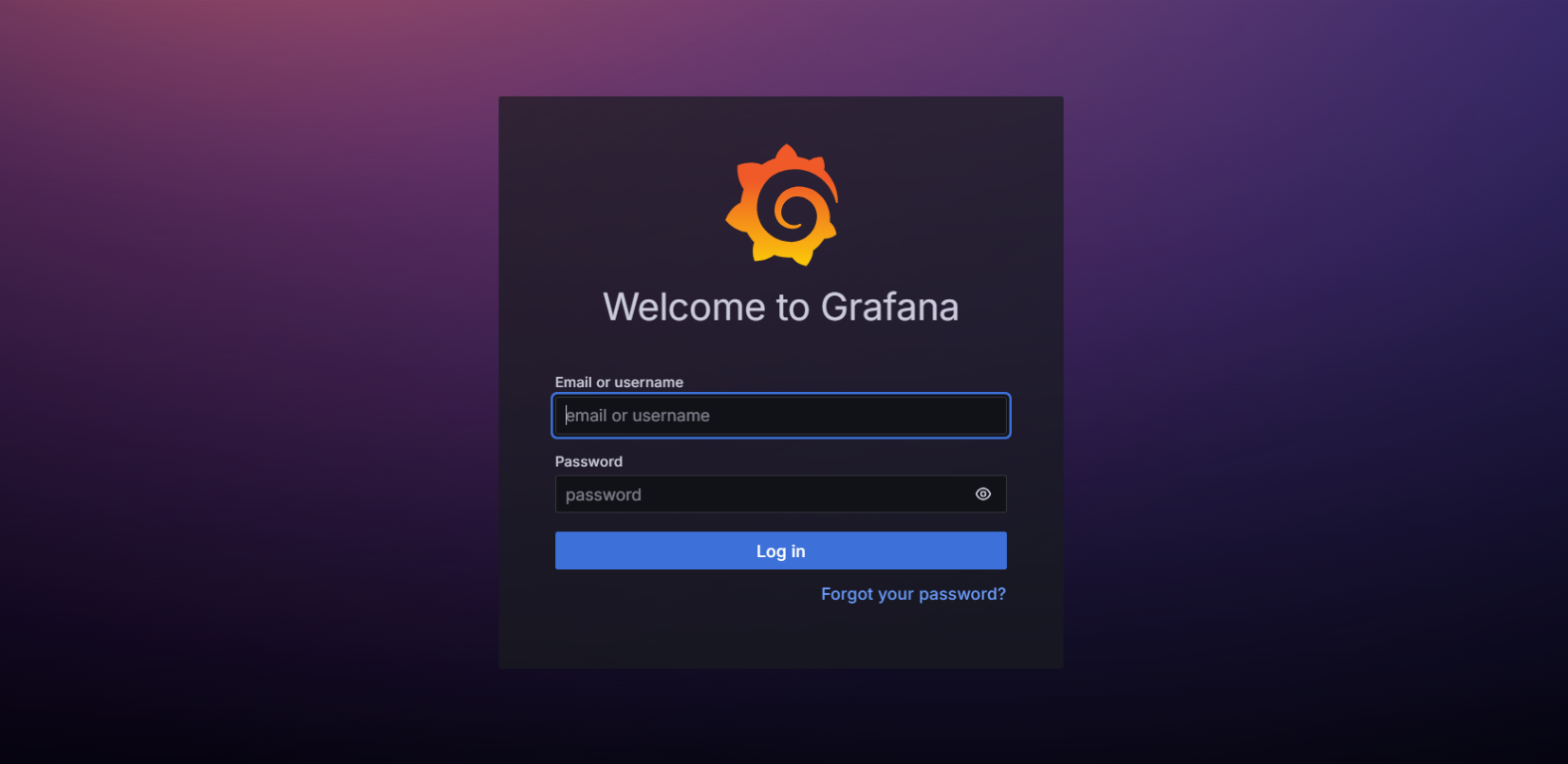
* Now click on the add rule.

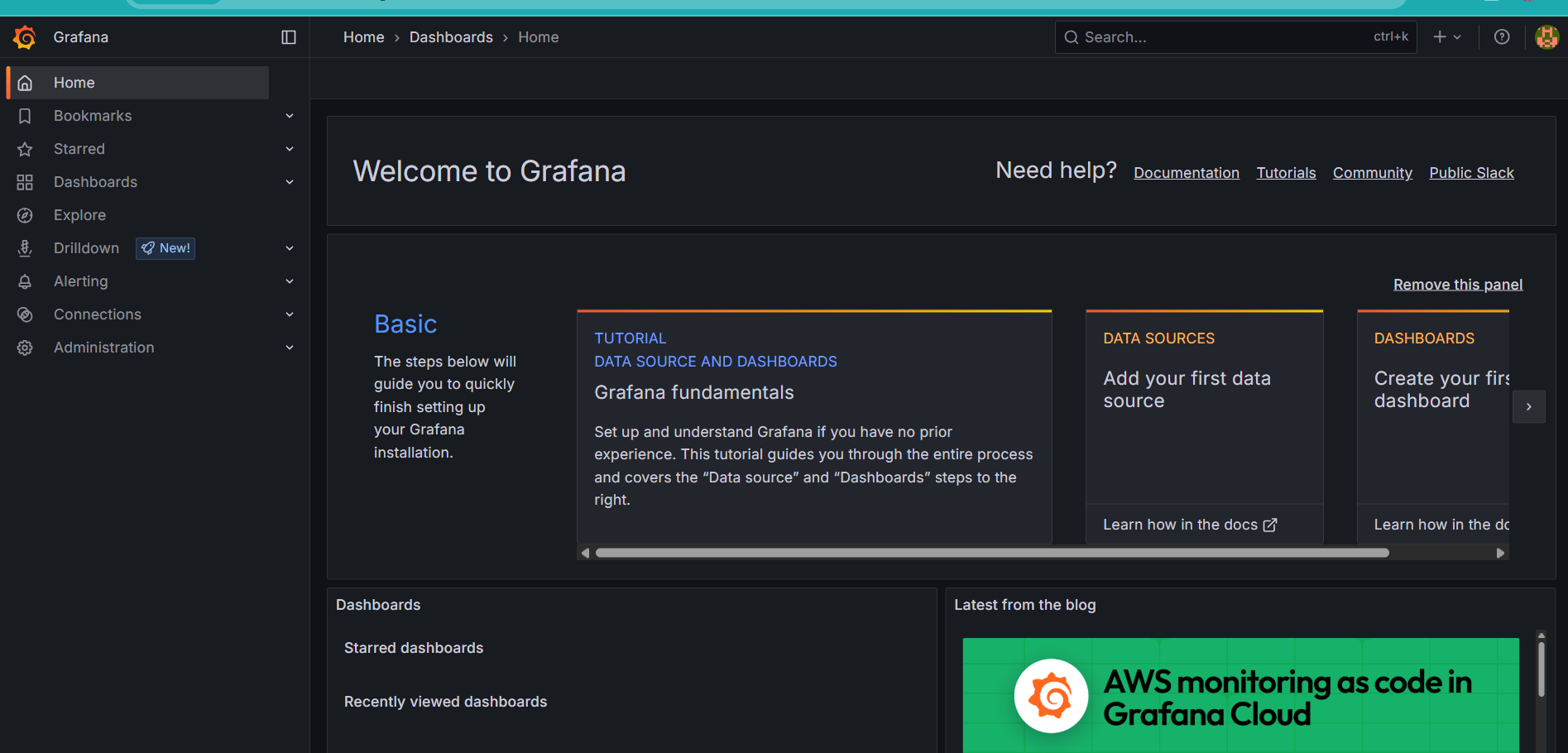
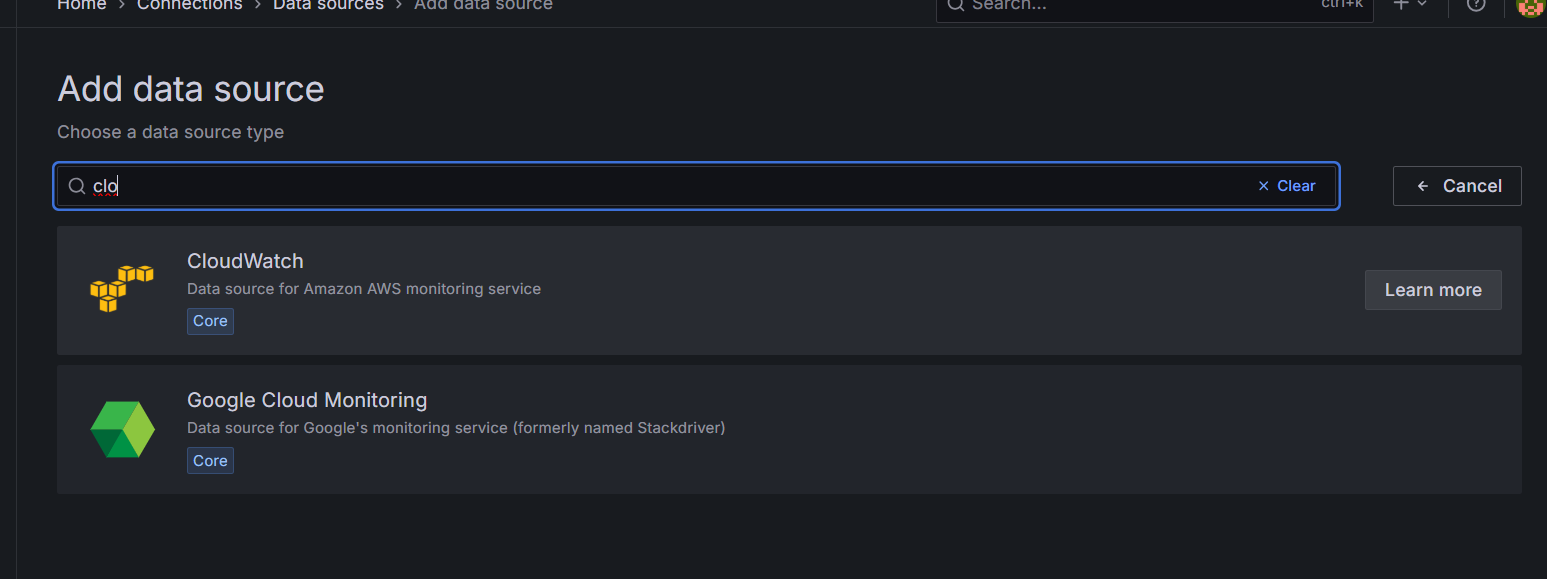
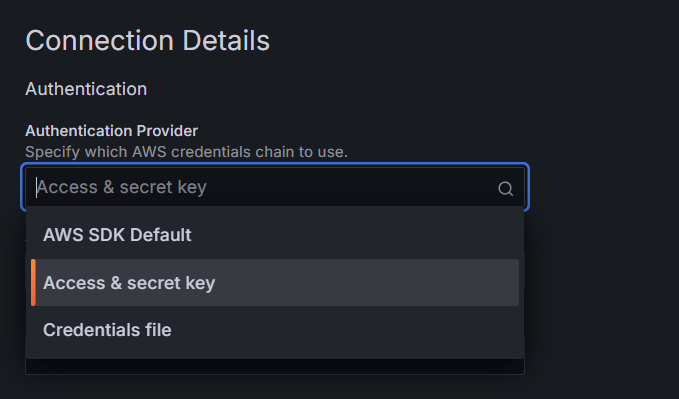


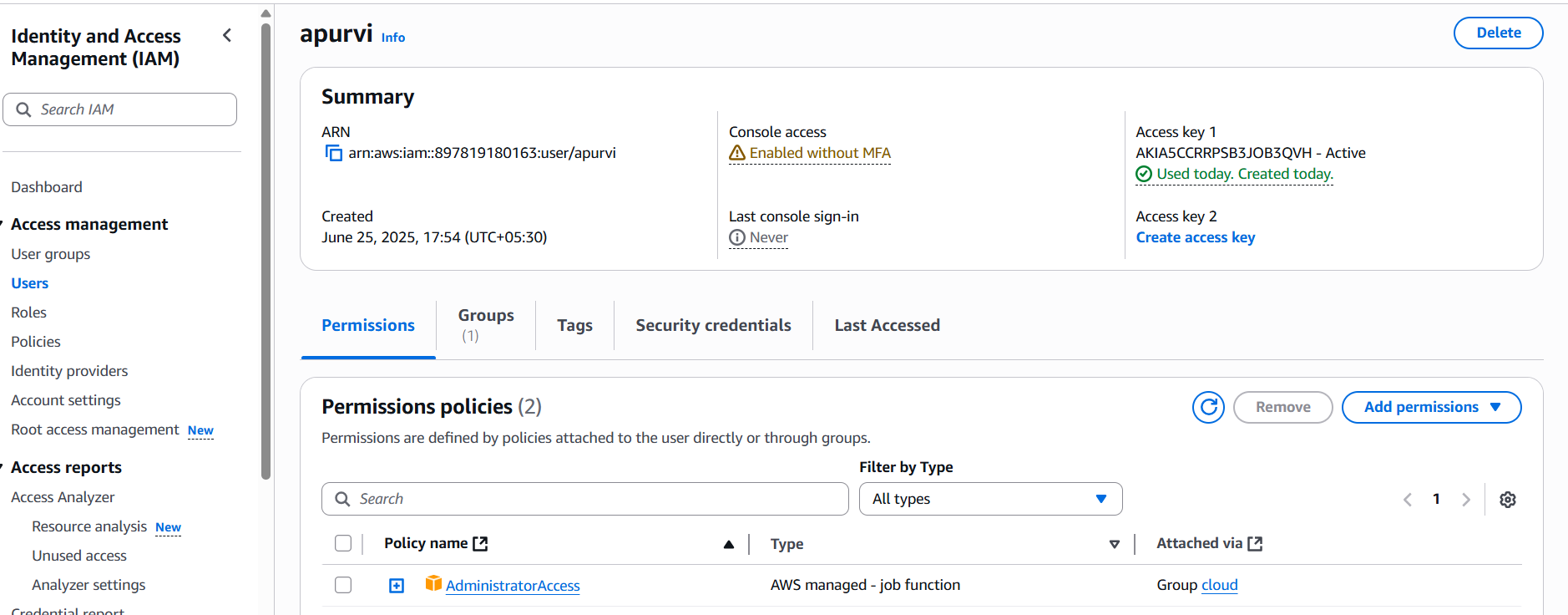
* Now click on save rules.
* Now copy your public IP and place it into new tab and add the port range(3000).

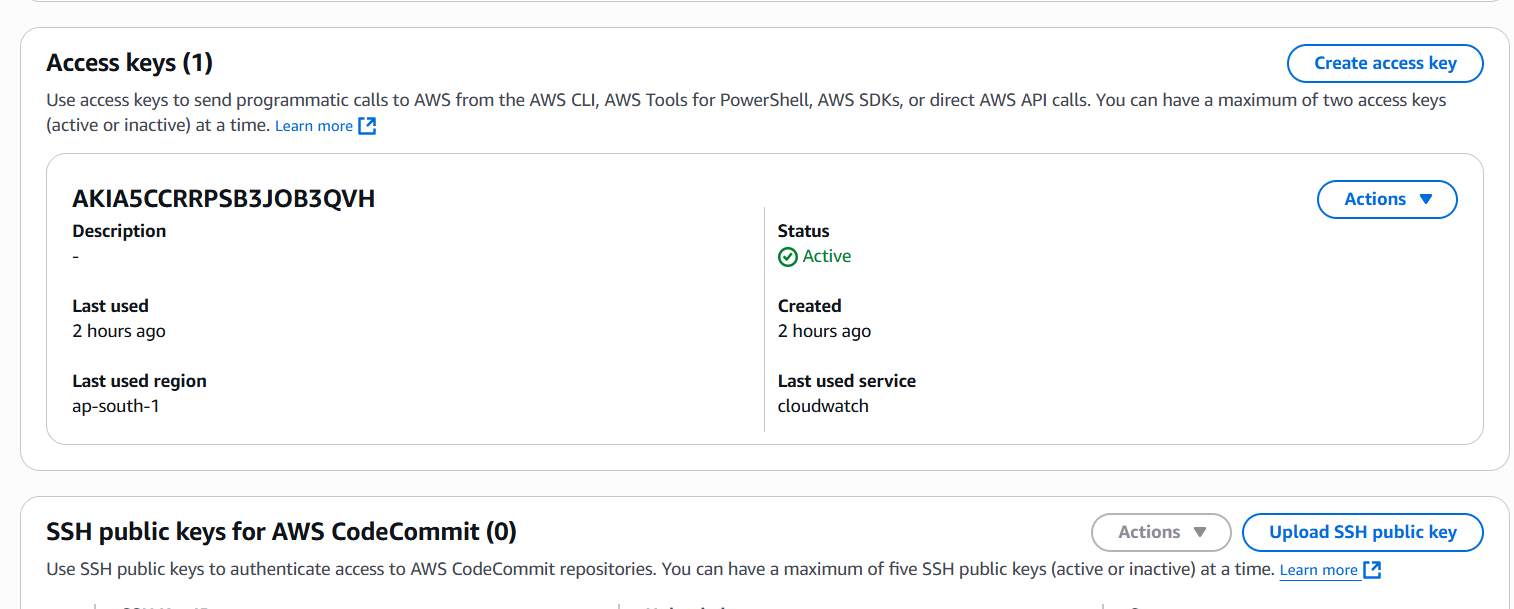
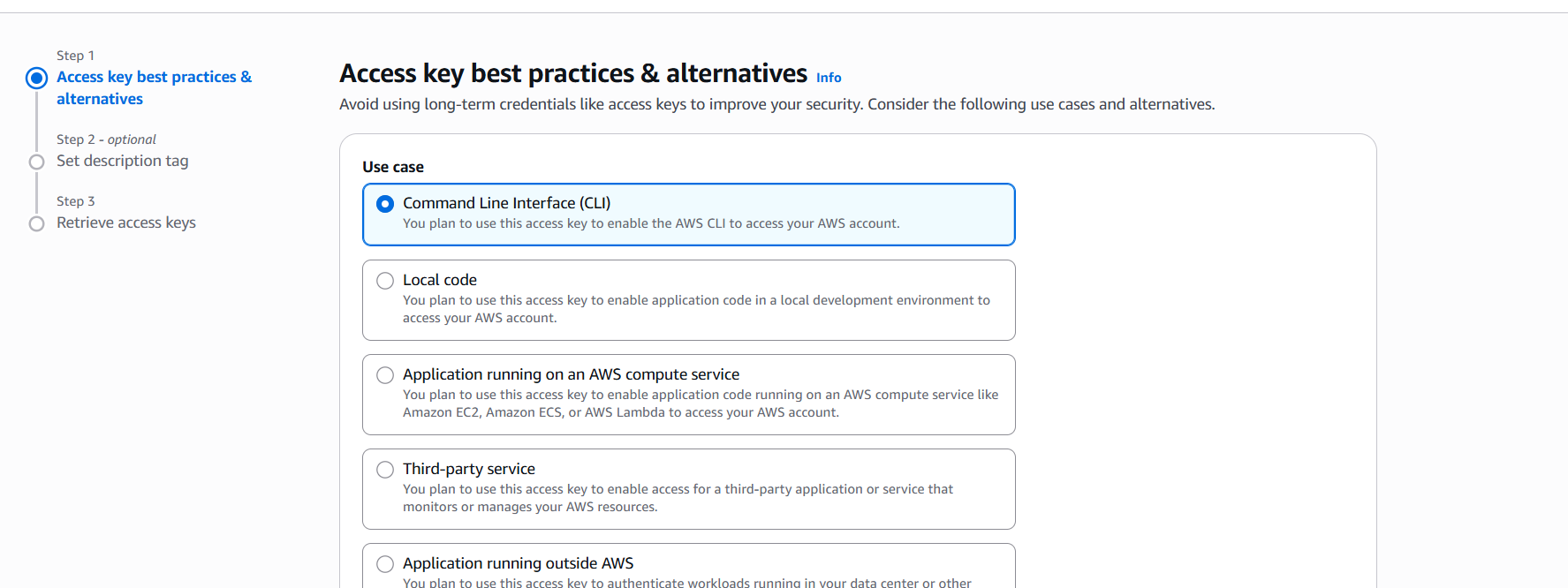


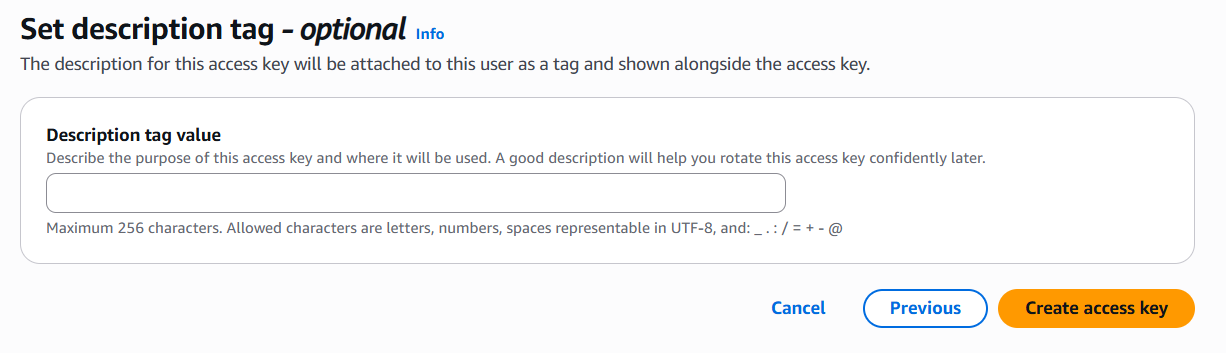
* Now your grafana will open.



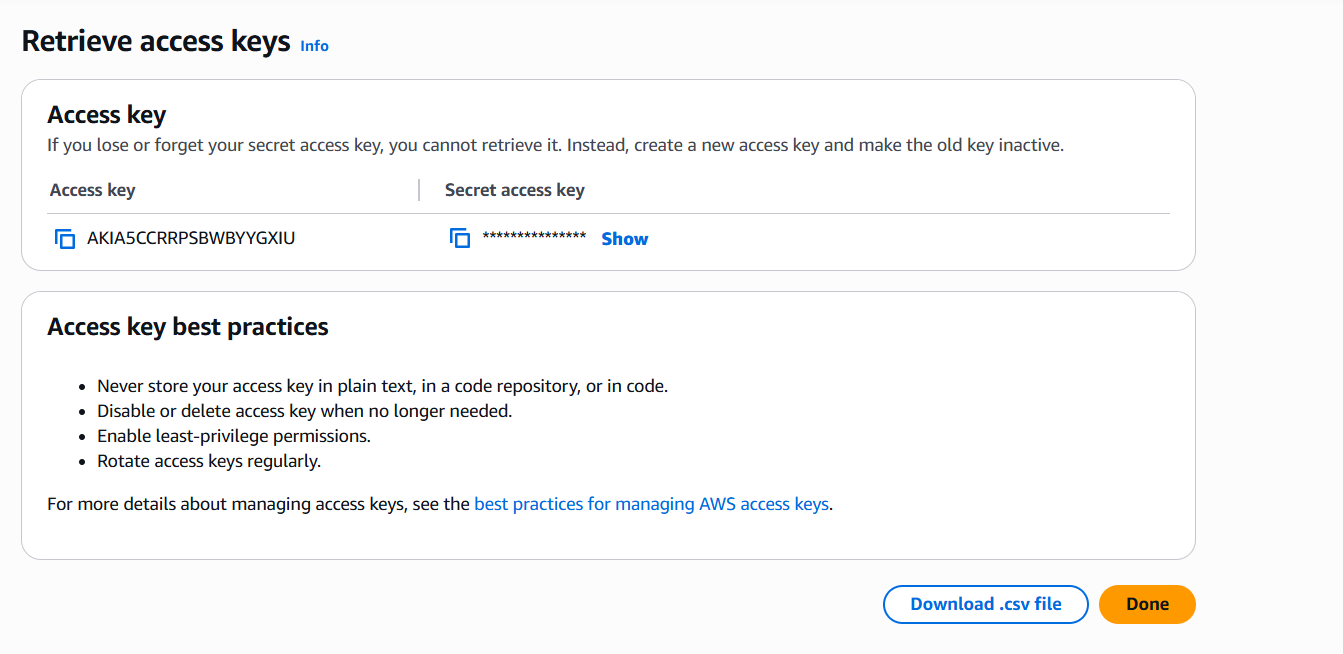
* Now in username write it admin and in password write in admin.
* Your grafana will open.
* Now go to connections and data source.
* Click on add data source.
* Search cloud watch and open it.
* In connection details :
* Now go to the IAM if you are a iam user and go to your user id if you are a root user then make your own user.



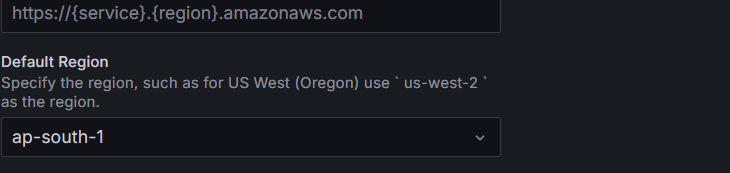
* Now go to security credentials.
* Click on create access key.



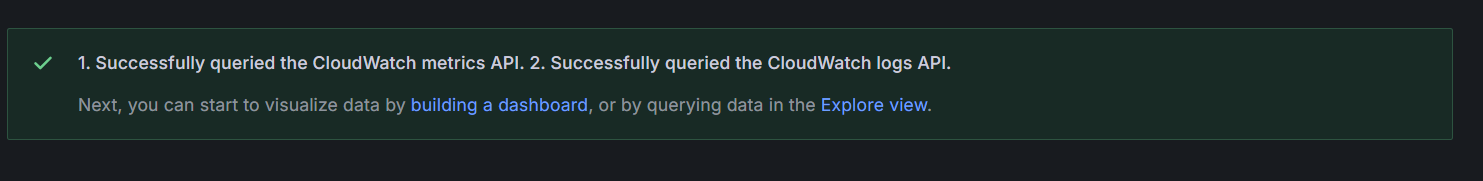
* Click on create access key.

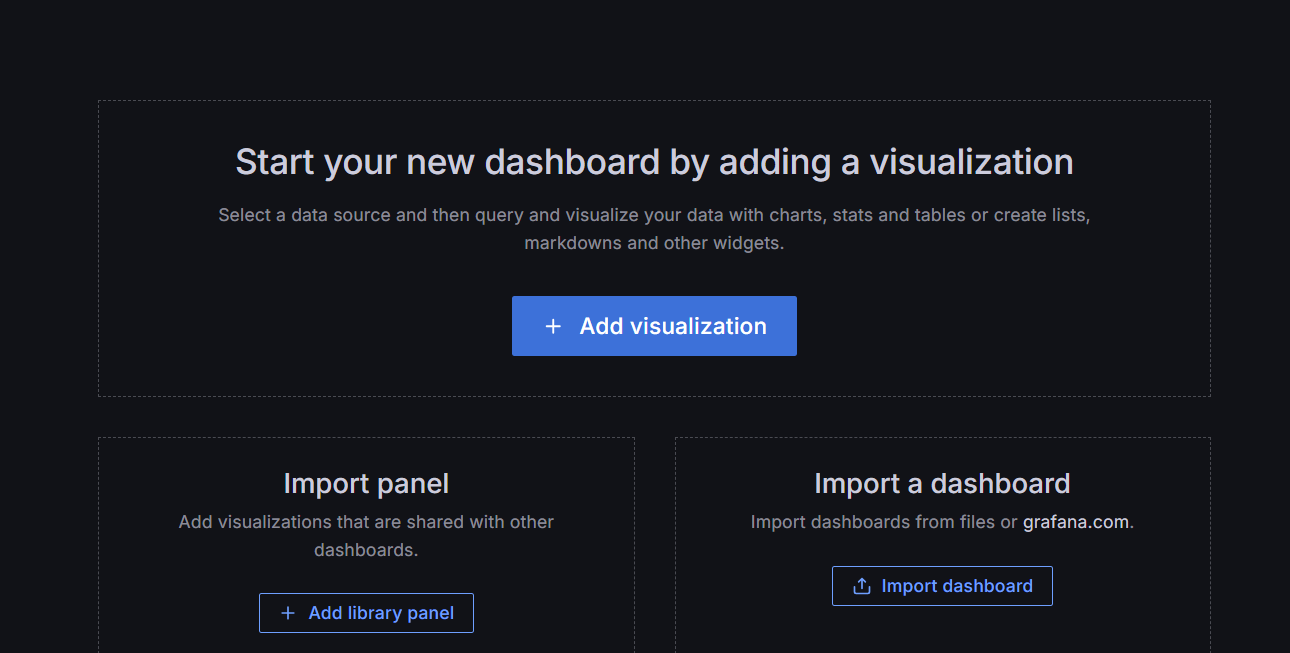
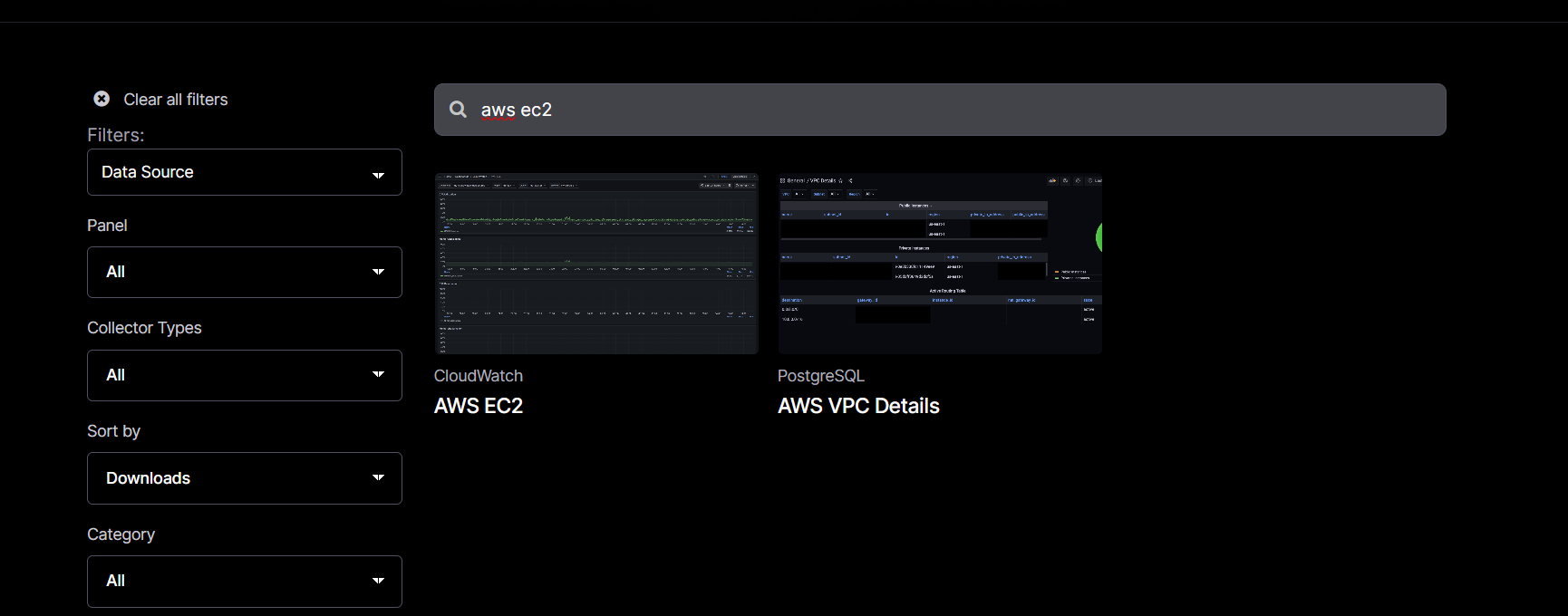


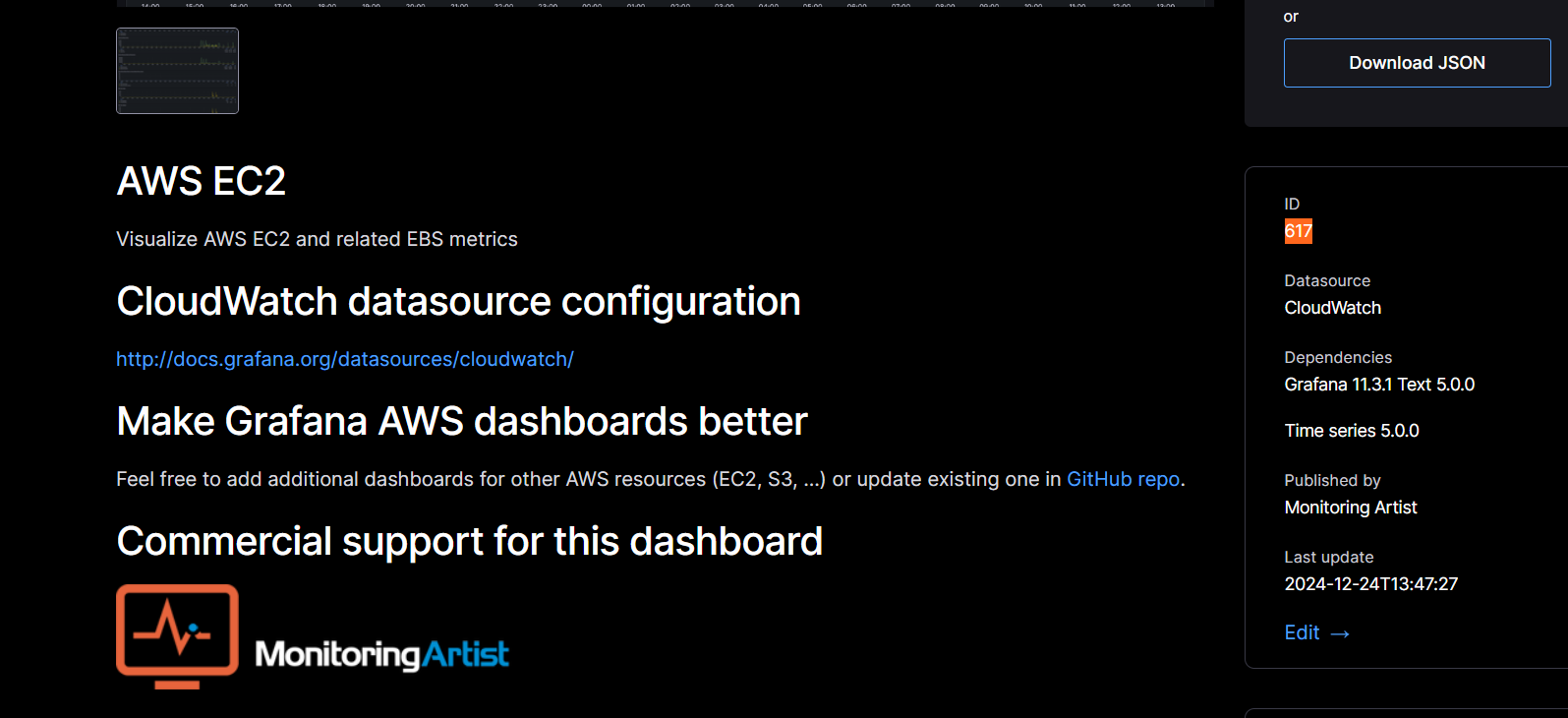
* Your private and access key will be generated now copy it and paste it .
* In region choose ap-south1.

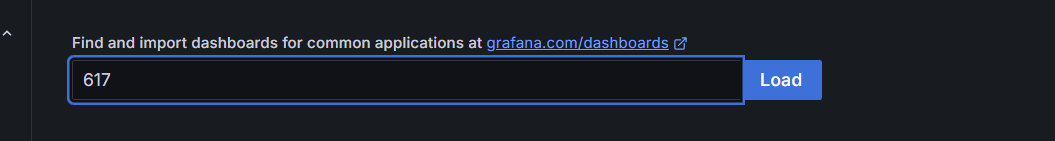


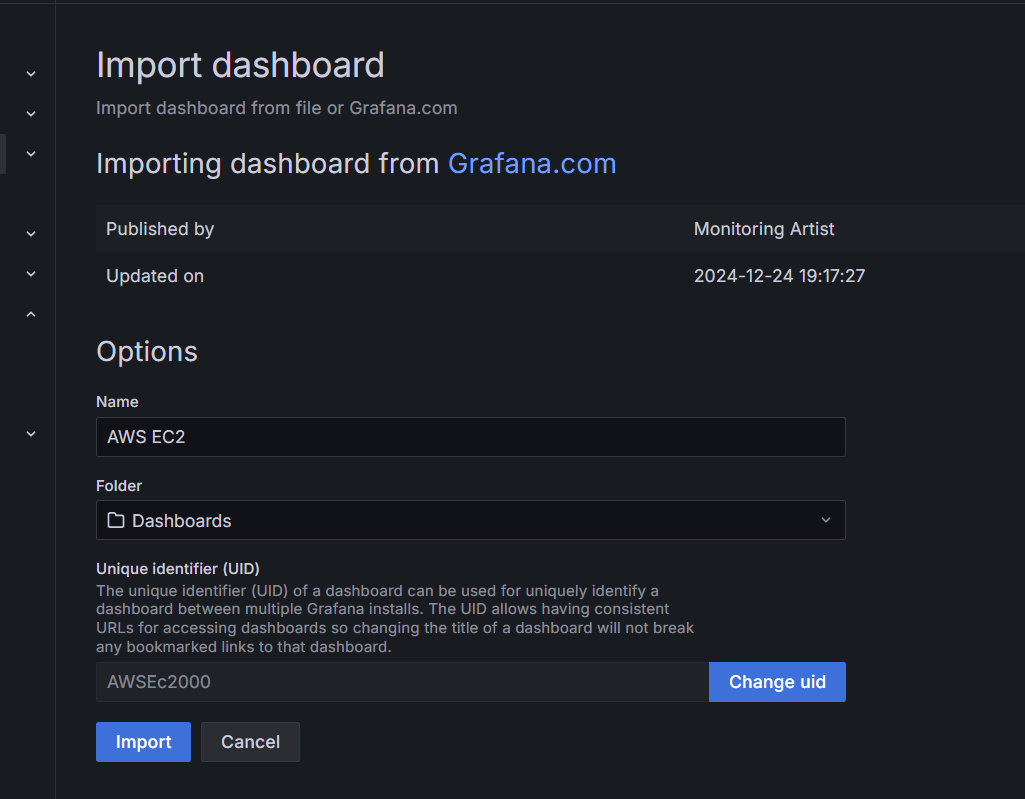
* Now click on save and test.



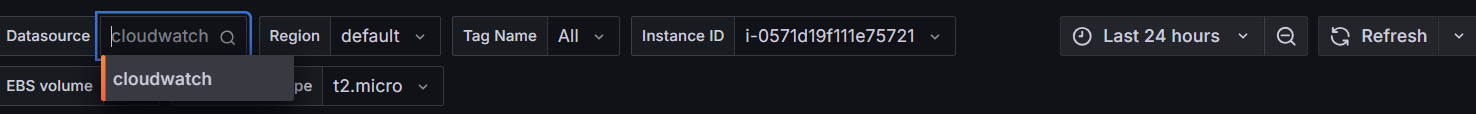
* Now go to home and click on dashboard.
* Click on import a dashboard.
* Click on granafana.com/dashboards.
* Search aws ec2
* Click on aws ec2 and open it .



* Now copy the id number and place it in the dashboard.

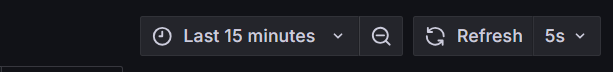


* Import the dashboard.
* Now in your dashboard :

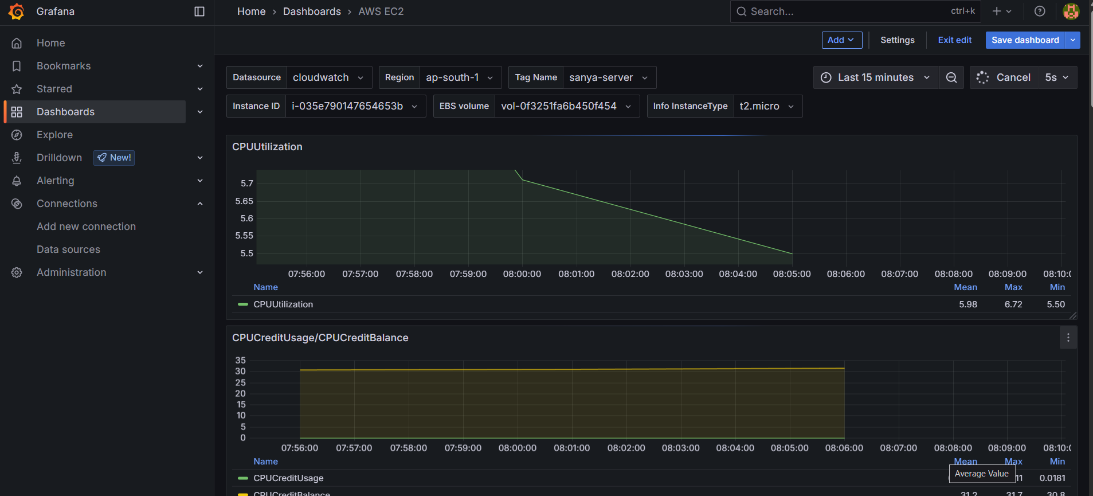
 Datasource: cloudwatch

Region: ap-south1

Tag-name: your instance name



* Choose 5secs and last 15 minutes



* Your graph will appear on screen