

Monitoring High CPU Utilization with Grafana on a Linux Server in AWS



**TABLE OF CONTENTS**

1. Introduction

2. Grafana

3. Steps for Deployment

Step 1: Set-up EC2 instance

Step 2: Install Grafana

Step 3: Login to Grafana

Step 4: Create policy for EC2 instance

Step 5: Set-up Grafana

4. Conclusion

5. References

**1. INTRODUCTION**

Effective monitoring and visualization of server performance metrics are essential for maintaining system health and efficiency. This project focuses on integrating Grafana with a Linux server to monitor high CPU utilization.

Grafana is an open-source platform for real-time monitoring and visualization. Prometheus is a monitoring and alerting toolkit optimized for reliability. By combining these tools, we can create a robust solution to monitor and visualize CPU usage, enabling proactive management of server resources.

This project involves setting up an AWS EC2 instance, installing Prometheus and Node Exporter for data collection, configuring Grafana for data visualization, and creating a custom dashboard to track CPU utilization. This integration helps identify performance bottlenecks and ensures efficient resource utilization, providing valuable insights into server performance.





**2. GRAFANA**

Grafana is an open-source platform for monitoring and visualizing metrics from various data sources. It allows users to create interactive, real-time dashboards that provide insights into system performance, application health, and business metrics.

**Key Features:**

**1. Multi-Source Data Integration:** Supports numerous data sources like Prometheus, InfluxDB, Elasticsearch, MySQL, and more.

**2. Customizable Dashboards:** Offers a wide range of visualizations including graphs, heatmaps, and tables.

**3. Real-Time Monitoring:** Displays live data and provides alerting capabilities to notify users of important events.

**4. User Management:** Supports user authentication and role-based access control, making it suitable for team use.

**5. Plugins and Extensibility:** Allows the addition of new data sources and visualizations through plugins.

**6. Open-Source:** Benefits from a large community that contributes to its development and support.

Grafana is essential for creating detailed, real-time dashboards that help in proactive system and application monitoring.

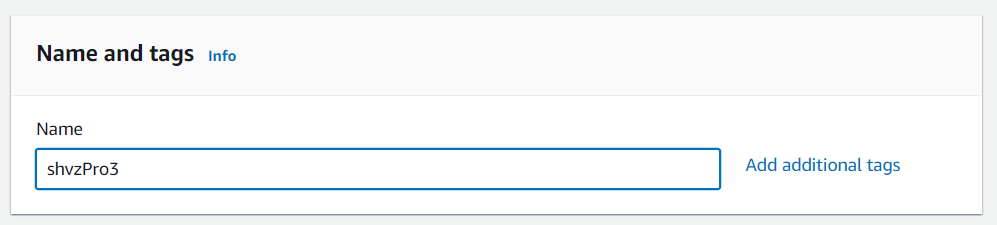
**3. STEPS FOR DEPLOYMENT**

**Step 1:**

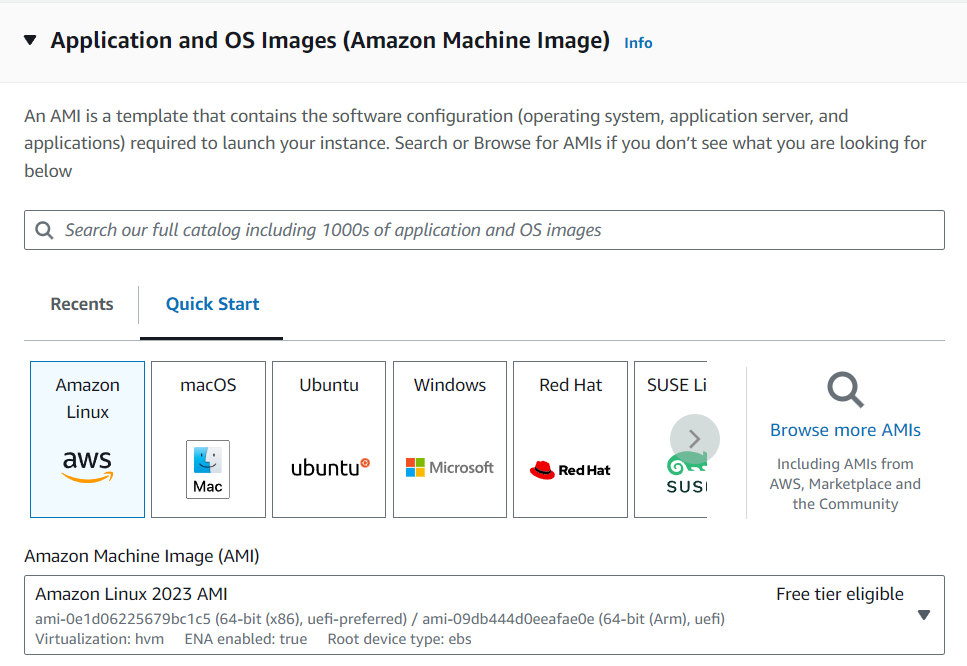
**Set Up AWS EC2 Instance**

1. **Launch an EC2 Instance**

* Log in to the AWS Management Console.
* Navigate to the EC2 Dashboard and click "**Launch Instance."**
* Choose an Amazon Machine Image (AMI), such as **Amazon Linux 2 AMI**.

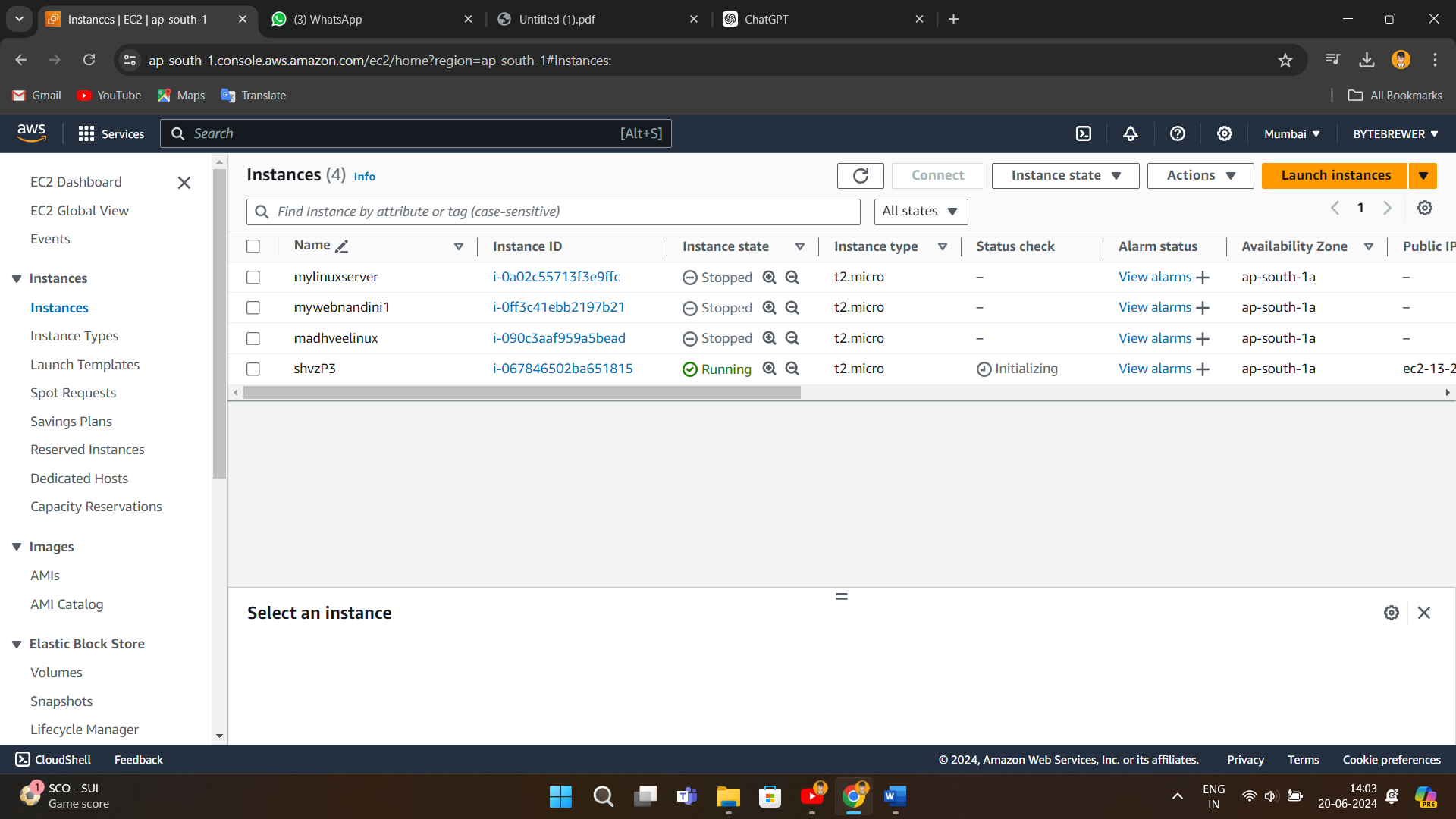


shvzP3



* Select an instance type, such as **t2.micro** (eligible for the free tier).
* Create a new keypair **(.ppk)** [for PuTTy]
* Configure instance details, add storage, and configure security group **(allow SSH (port 22) and Grafana (port 3000) ports).**
* Review and launch the instance.

1. **Connect to the EC2 Instance**

* Click on the instance that you created and copy the *Public IPv4 address*
* Connect to your instance by pasting the address and uploading the key (.ppk) into the PuTTy.
* Type ‘ec2-user’ and hit Enter to login



**Step 2:**

**Install Grafana**

* Run the following commands

a: Change to Root user: **sudo su**

b: The installation is via yum repository.

Create a **grafana.repo** under **/etc/yum.repos.d/**

**vi grafana.repo**

c: Put below content in grafana.repo and save it.

[grafana]

name=grafana

baseurl=https://packages.grafana.com/oss/rpm

repo\_gpgcheck=1

enabled=1

gpgcheck=1

gpgkey=https://packages.grafana.com/gpg.key

sslverify=1

sslcacert=/etc/pki/tls/certs/ca-bundle.crt

d: Once done with the above steps, It's time to install Grafana.

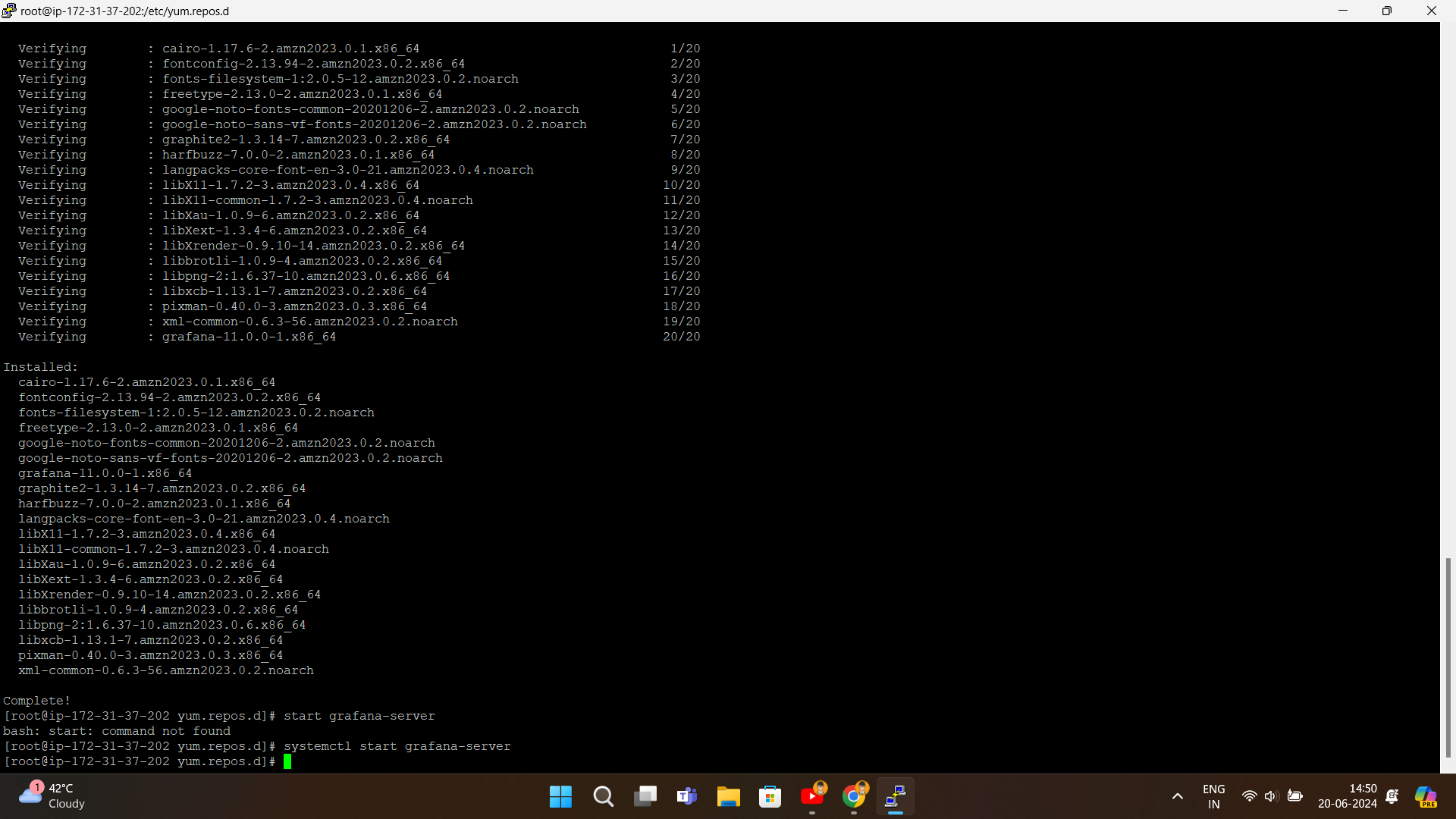
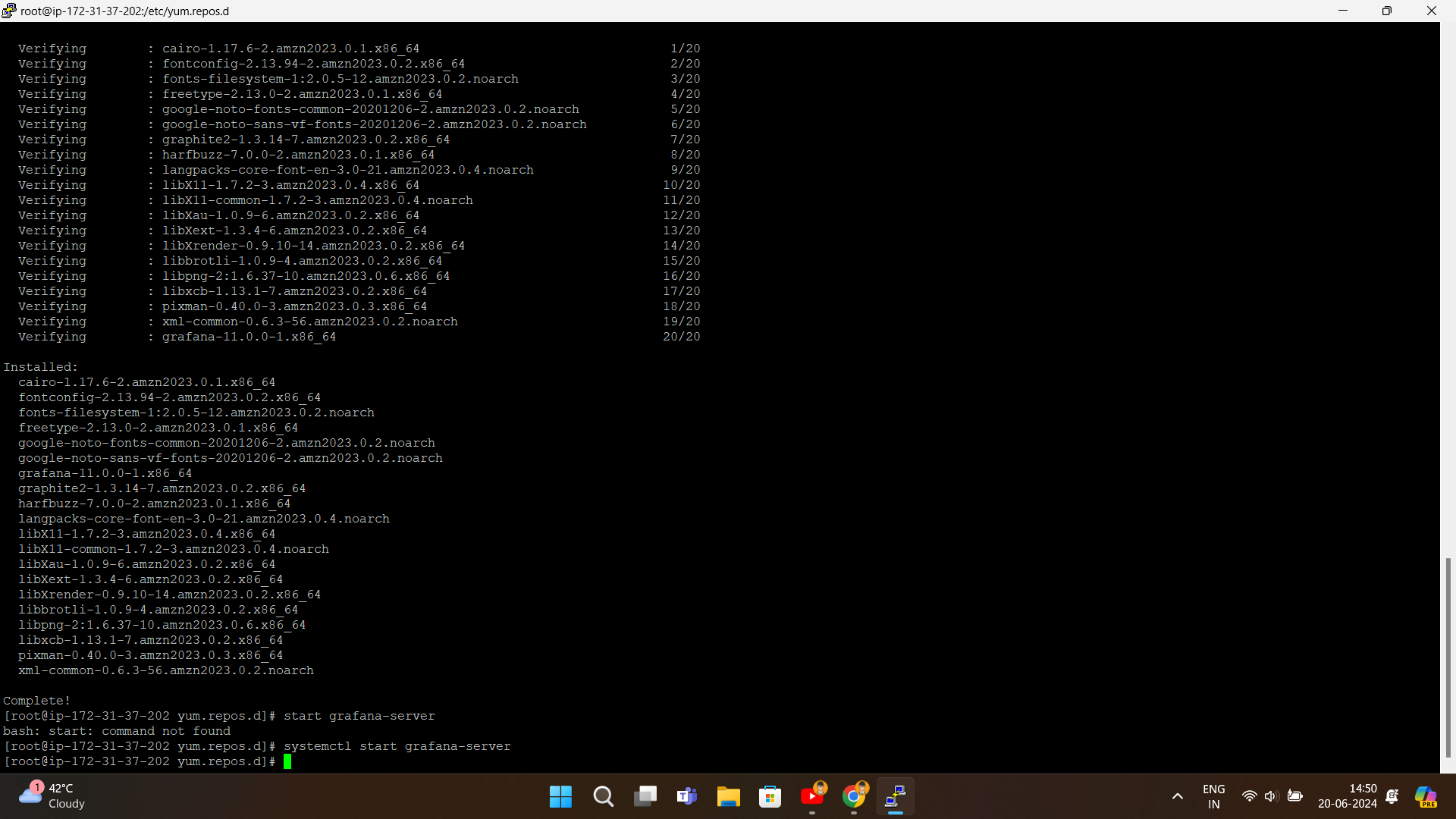
Type “**sudo yum -y install Grafana**”

e: Start the server -

**systemctl start grafana-server**

**systemctl status grafana-server**





***Grafana is successfully installed !!!***

**Step 3:**

**Login to Grafana**

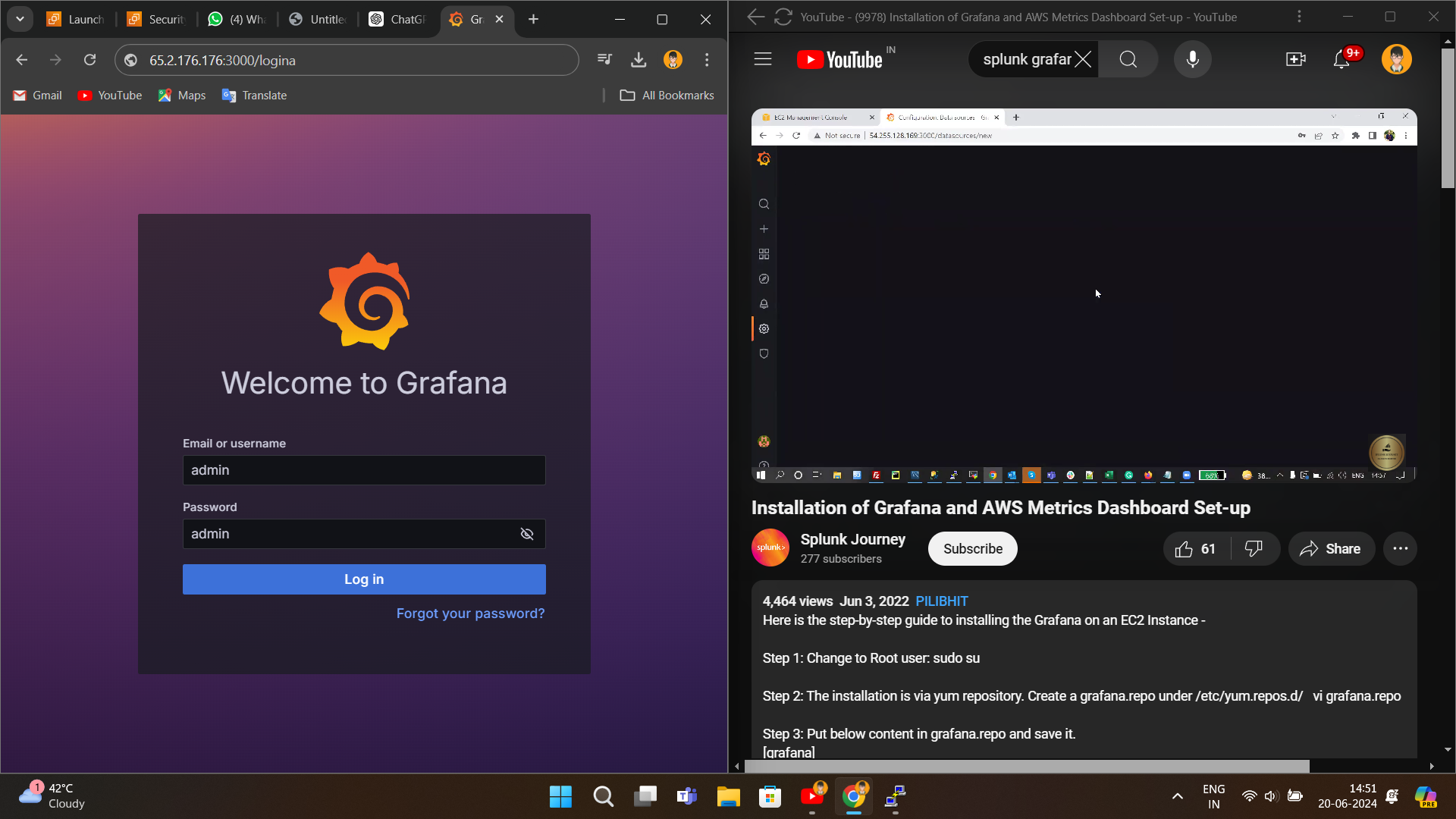
* Copy your instance’s *Public iPv4 address* and add **:300/login**

 **Ex-** 65.2.176.176 🡪 65.2.176.176:300/login

* Hit the enter button and you’ll be redirected to the Grafana Login page

*Userame:* **admin**

*Password:* **admin**



* After login, you can change the username and password

**Step 4:**

**Create a policy for EC2 instance**

* Go to **IAM > Policies > Create policy**
* Click on **JSON** and paste the Policy given below

{

"Version": "2012-10-17",

"Statement": [

{

"Sid": "VisualEditor0",

"Effect": "Allow",

"Action": [

"ec2:DescribeInstances",

"cloudwatch:GetMetricData",

"ec2:DescribeTags",

"ec2:DescribeRegions",

"cloudwatch:GetMetricStatistics",

"cloudwatch:ListMetrics"

],

"Resource": "\*"

},

{

"Sid": "AllowReadingTagsInstancesRegionsFromEC2",

"Effect":"Allow",

"Action": ["ec2:DescribeTags","ec2:DescribeInstances","ec2:DescribeRegions"],

"Resource":"\*"

},

{

"Sid": "AllowReadingResoucesForTags",

"Effect":"Allow",

"Action":"tag:GetResources",

"Resource":"\*"

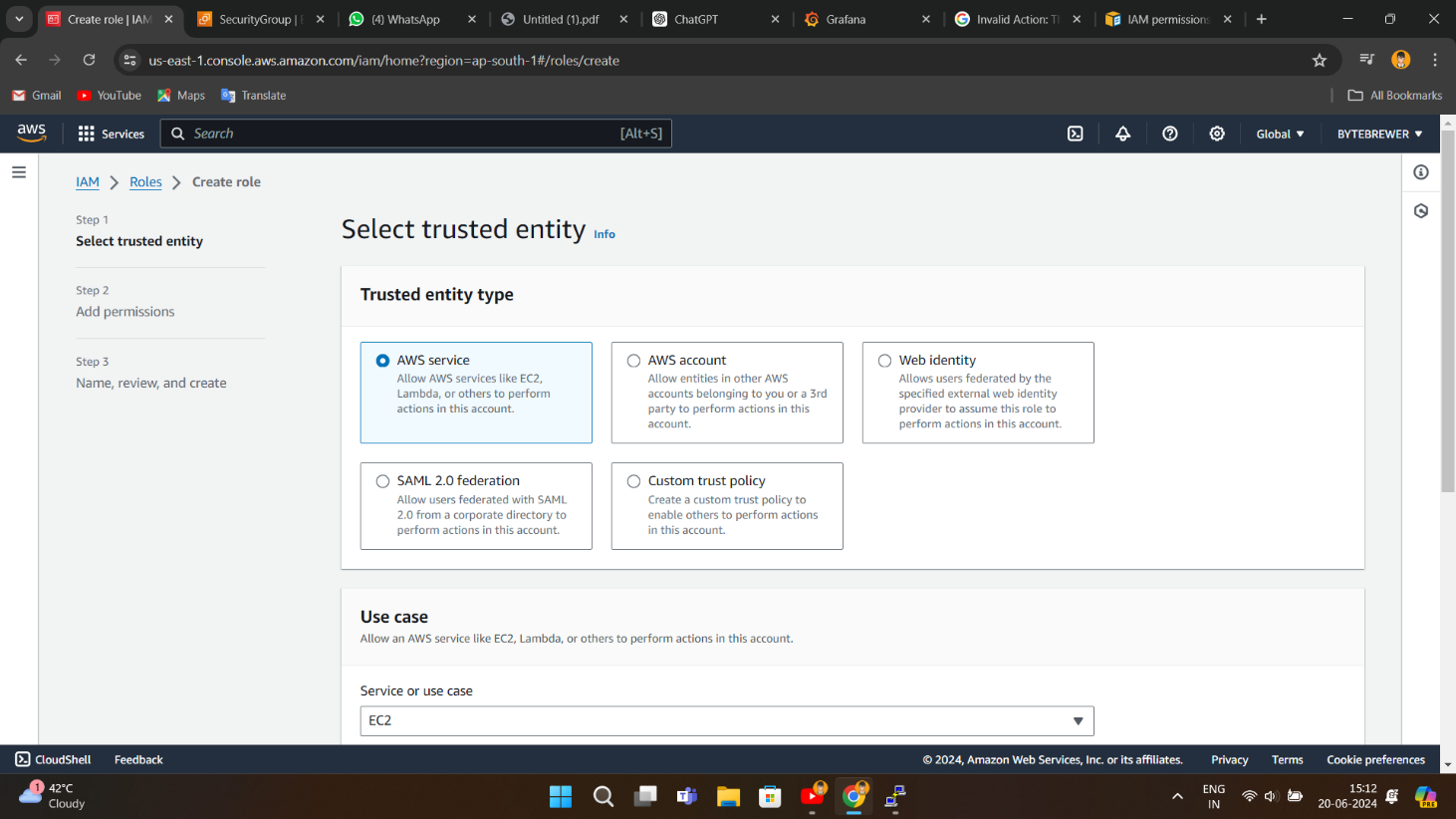
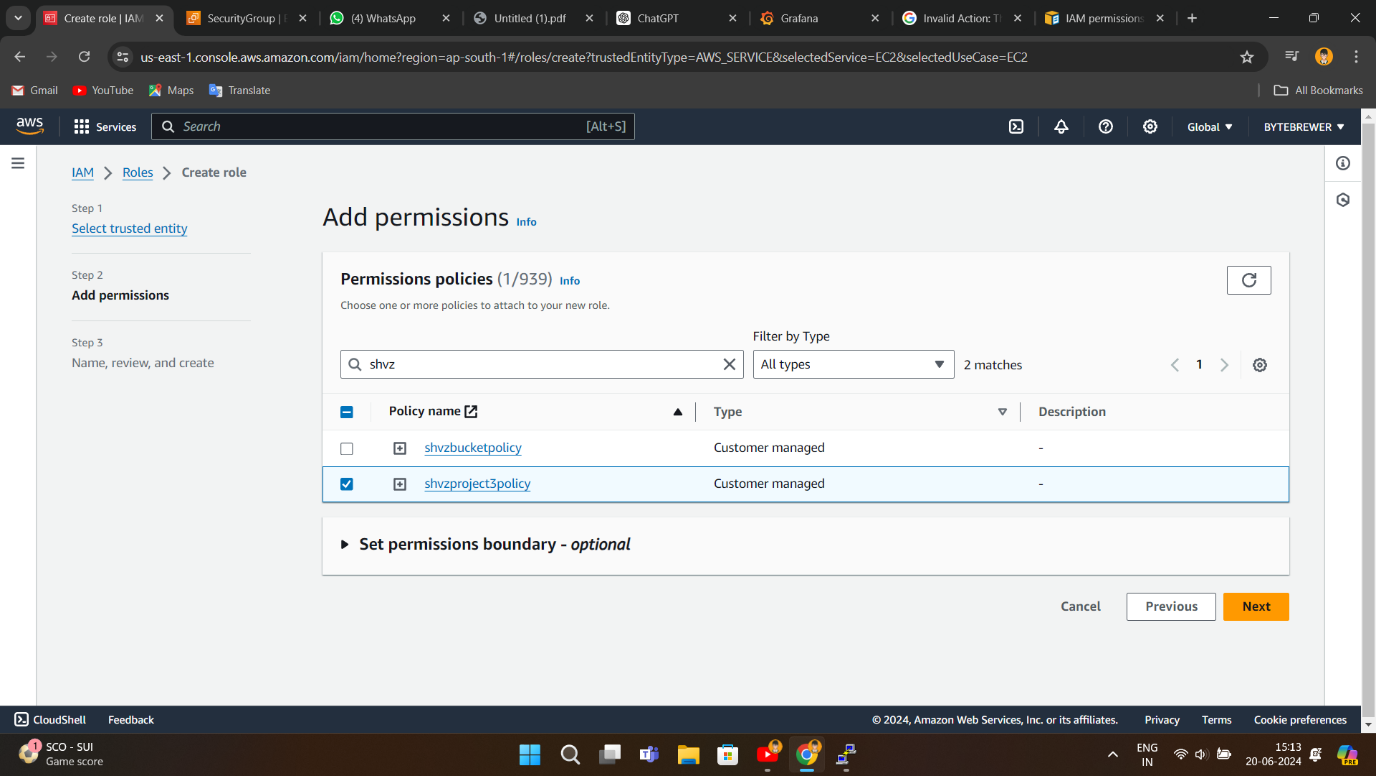
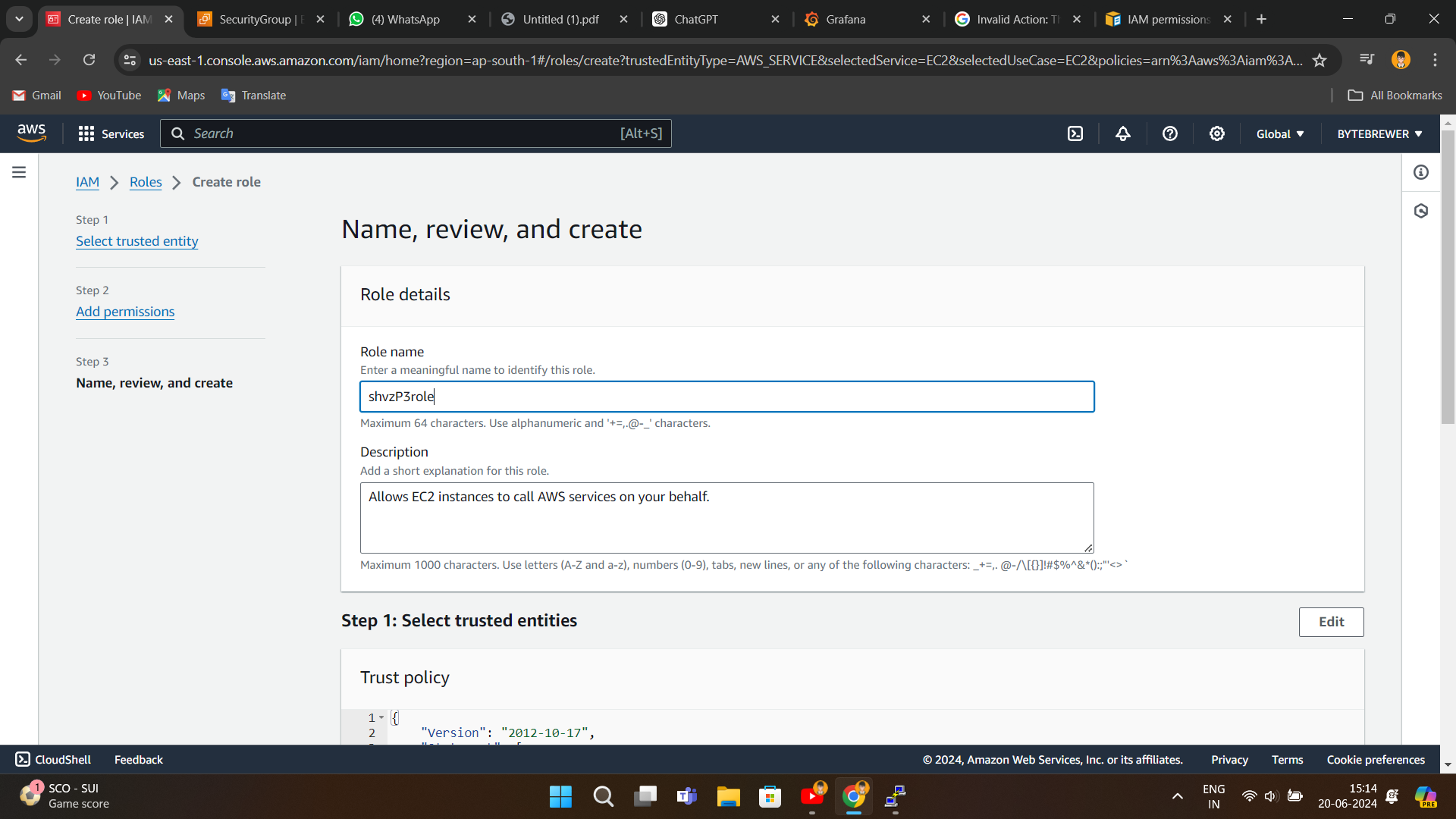
}

]

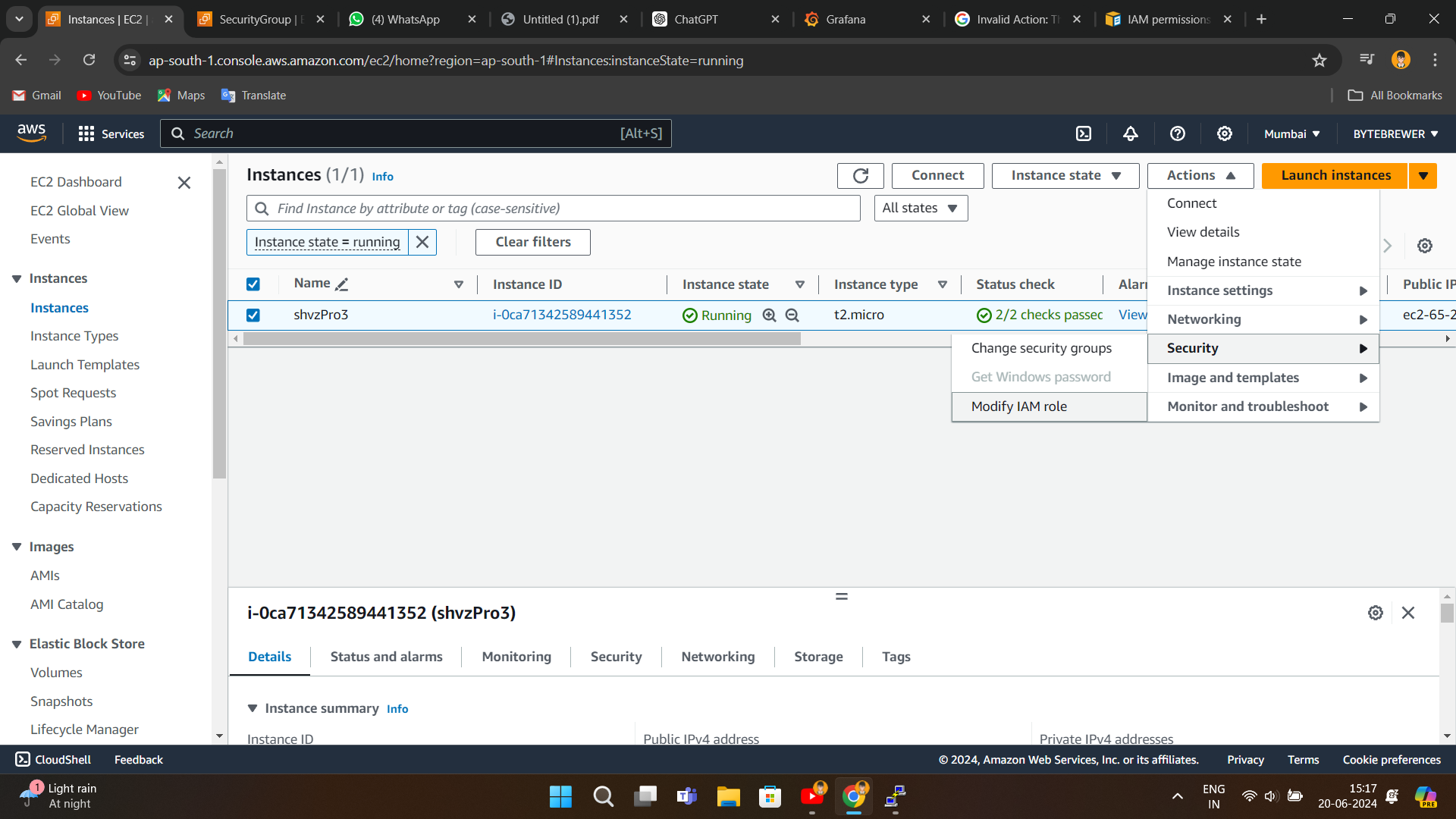
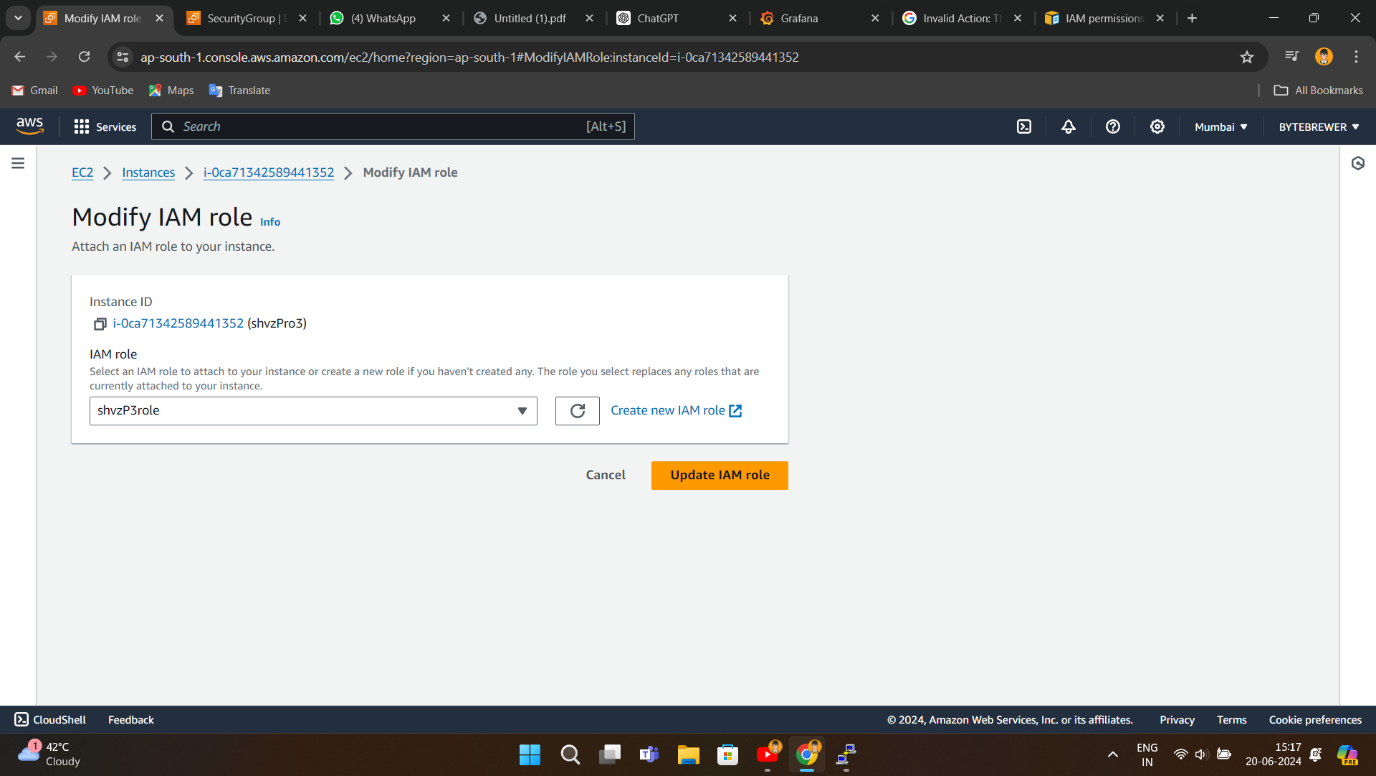
}

* Give a **name** to your policy and **save it.**

*Now attach this policy to a Role and to do this we need to create a role and attach this policy to it.*

* Go to **IAM > Role > Create role**
* **Attach the Policy that we created**
* **Give a name to the role and hit Create role button**

*Now go to instances tab and Modify IAM Role, to do this –*

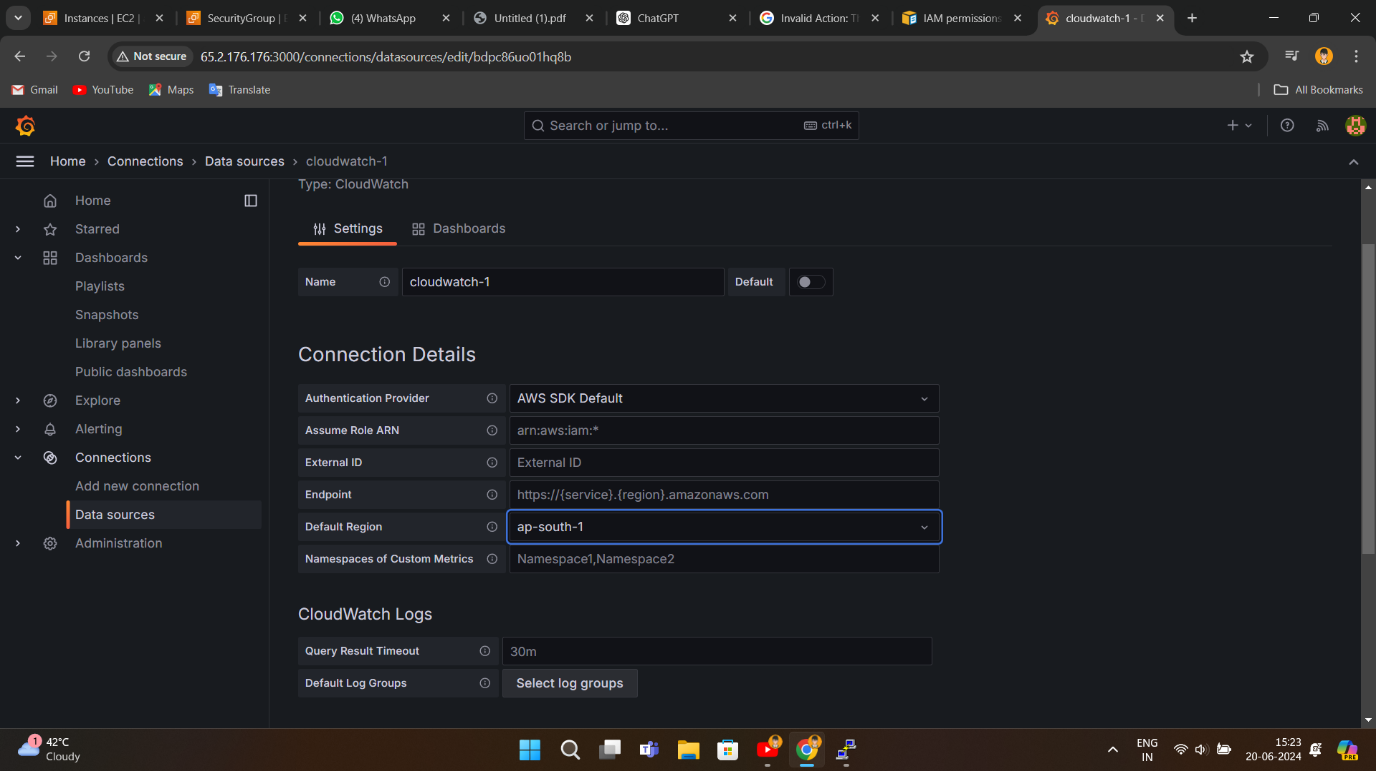
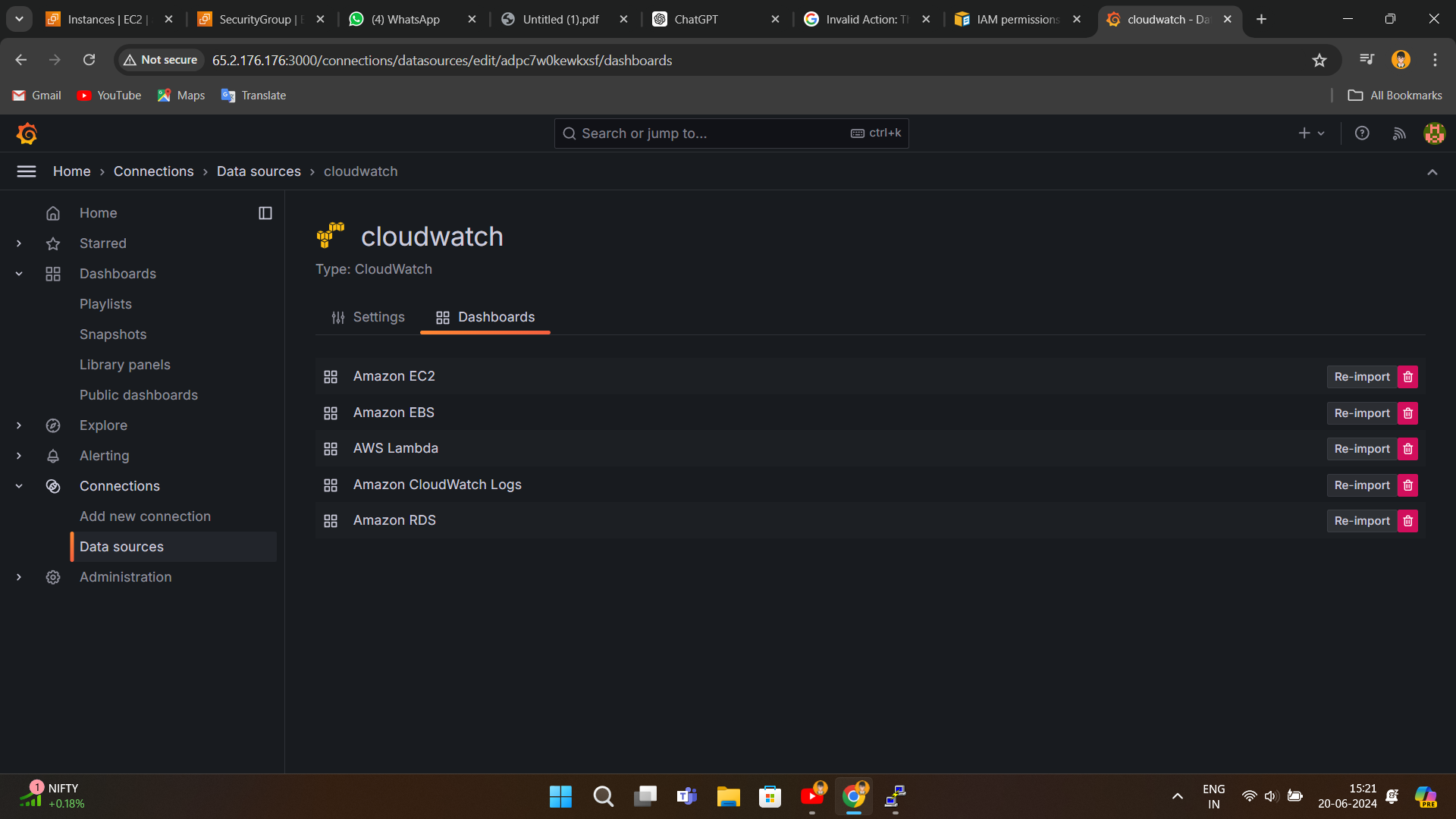
* *Select your instance* ***>******ACTIONS******>******Security******>******Modify IAM role***
* *Select the role we created and click on***Update IAM role**

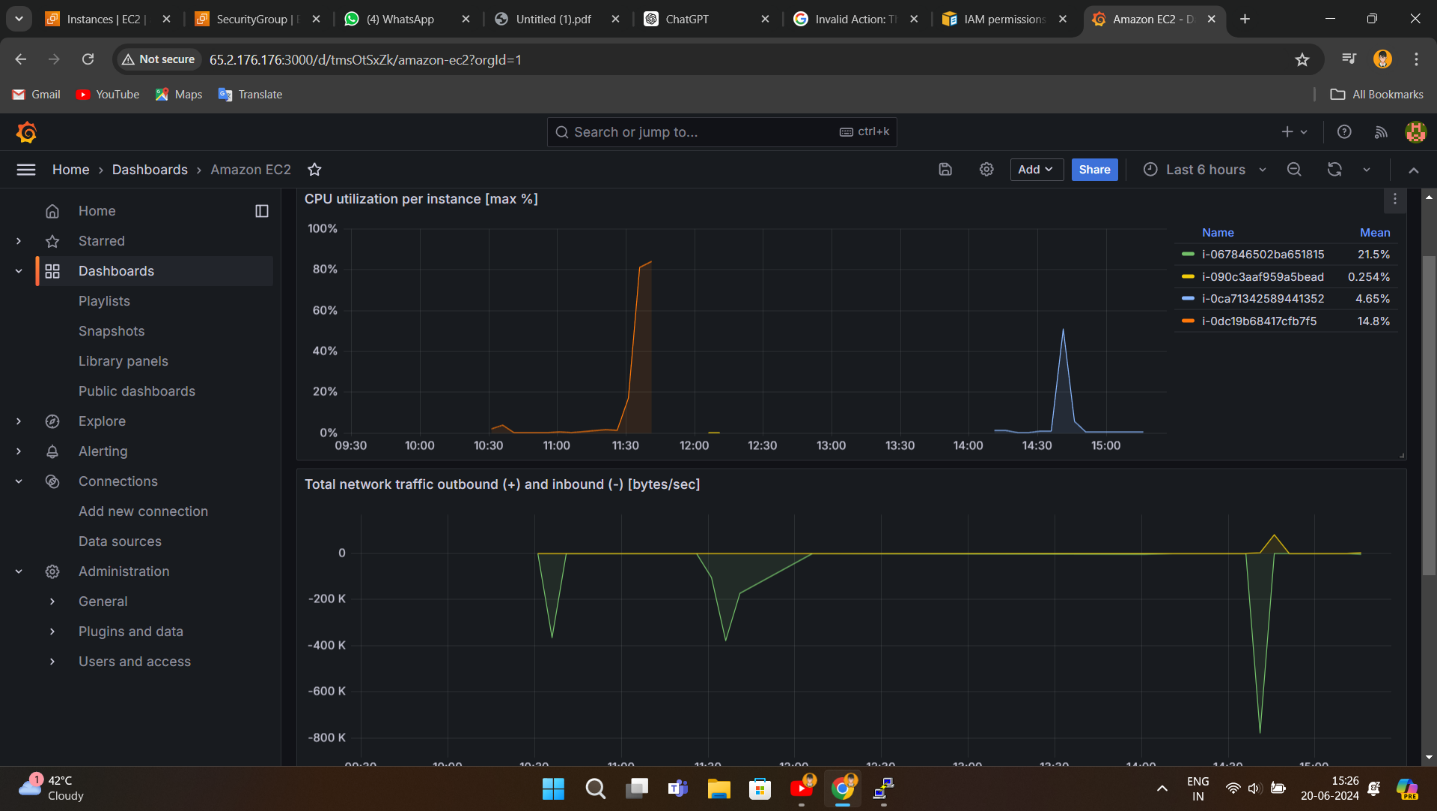
*All set, now we have add new connection (CLOUDWATCH) to Grafana and add dashboards*

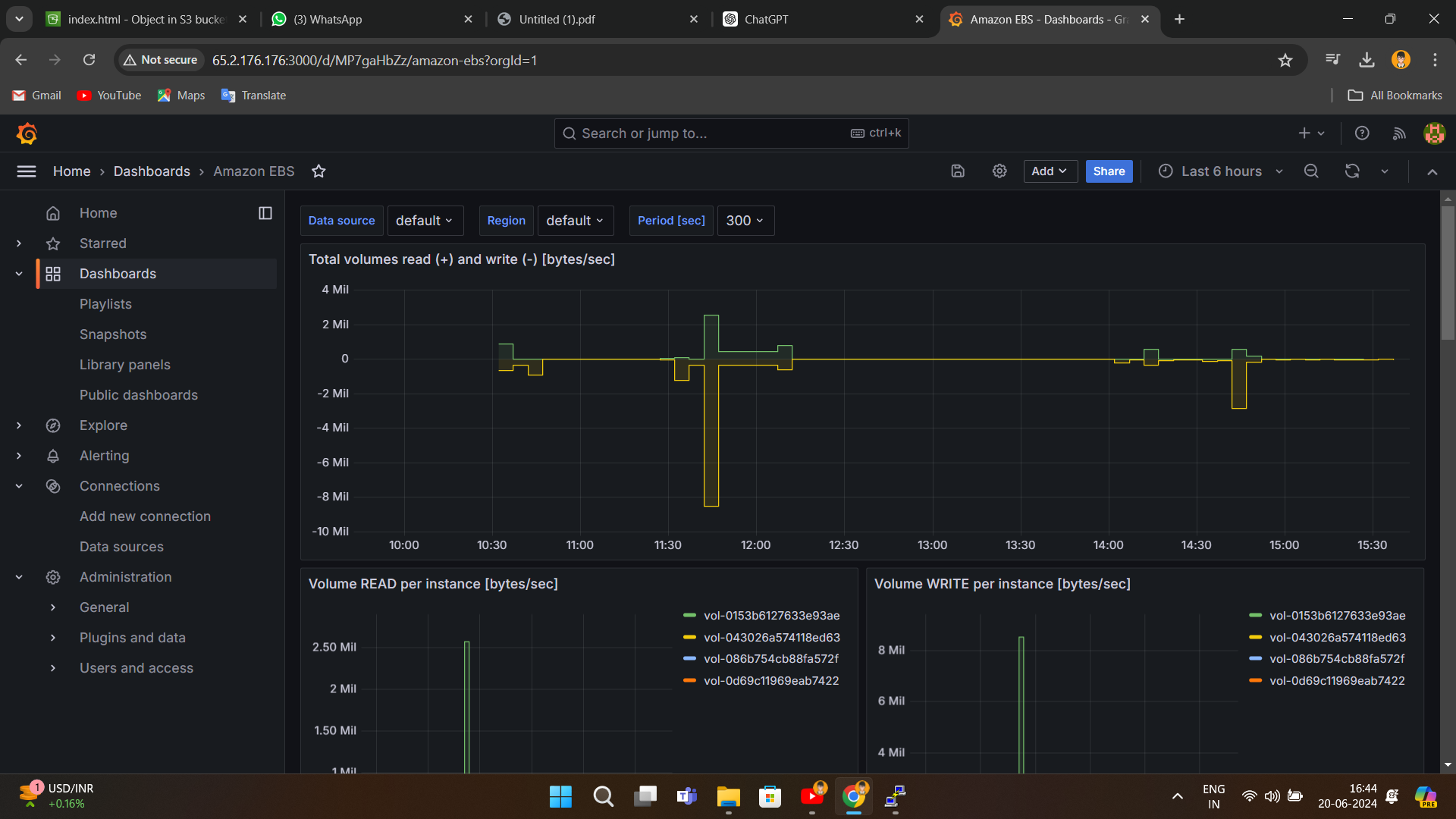
**Step 5:**

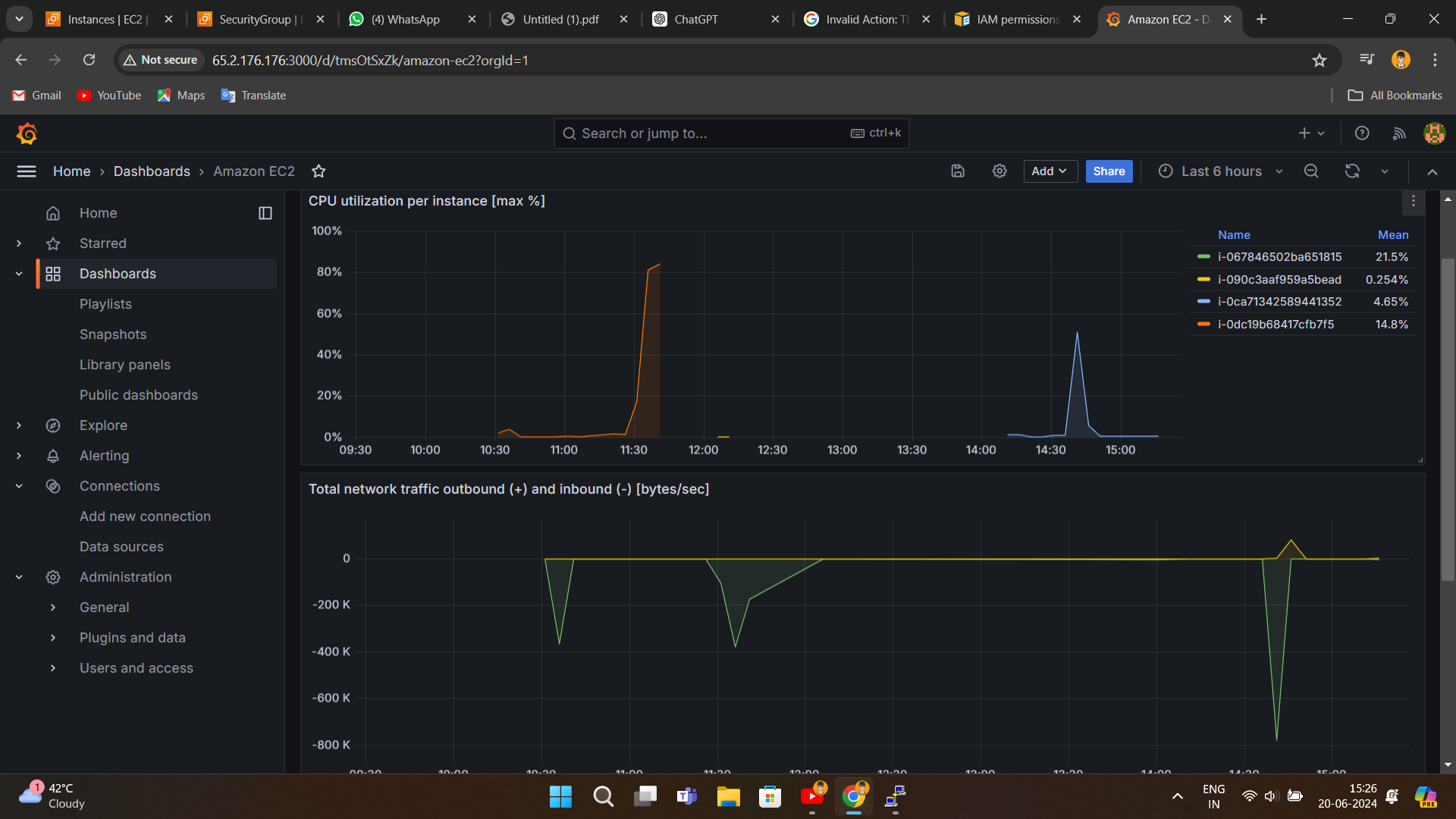
**Set-Up Grafana**

*We need to add a new connection to Grafana using Cloudwatch so that it can display the monitoring data. To do this –*

* *Go To* ***Grafana*** *>* ***Home*** *>* ***Connections*** *>* ***Add new connections > Cloudwatch*** *(search it)*
* **In Settings** > choose the region **ap-south-1 > Save/Test**
* **Dashboards** > import the dashboards

*Now check the Dashboard*

**



**4. CONCLUSION**

In this project, we successfully integrated Grafana with a Linux server and connected it to AWS Cloudwatch to monitor high CPU utilization and other critical metrics. By following the detailed steps provided, we configured Grafana to visualize performance data from Cloudwatch, enabling comprehensive and real-time monitoring of our server and AWS resources. This setup enhances our ability to manage and optimize system performance proactively, leveraging the power of Grafana's visualization capabilities combined with AWS Cloudwatch's robust monitoring features.

This integration demonstrates the efficiency and effectiveness of using Grafana and Cloudwatch for robust infrastructure monitoring and management.

**5. REFERENCES**

* [Grafana](https://grafana.com/docs/grafana-cloud/monitor-infrastructure/aws/)
* [Amazon Managed Grafana](https://aws.amazon.com/blogs/mt/amazon-managed-grafana-getting-started/)