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

Table of Content:

1. **What is Grafana?**
2. **How to Install Grafana**
    - **Create EC2 Instance**
    - **Install Grafana**
    - **Test Grafana**

3. Configure AWS cloudwatch dashboard

- **Create a new Policy**
- **Creating a Role**
- **Creating a User**
- **Attach the user to EC2 Instance**
- **Create your Dashboard**

4.Conclusion

1. What is Grafana?

Per Grafana “Grafana is a complete observability stack that allows you to monitor and analyze metrics, logs, and traces. It allows you to query, visualize, alert on and understand your data no matter where it is stored. Create, explore, and share beautiful dashboards with your team and foster a data-driven culture”. This is a very useful tool to visualize different logs, errors, or metrics from your instances.

2. How To Install Grafana

Step 1: Create EC2 Instance

We need to install Grafana on our server first, so let's provision an EC2 instance

- Go to EC2
- Create EC2
- Choose a public subnet

- Give it our Grafana security group

Now SSH into our instance

- Update your packages with

```
sudo yum update -y
```

We need to add a repository for grafana so our OS will know where it is.

```
sudo nano /etc/yum.repos.d/grafana.repo
```

Add the text below to the repo file. This will install the open-source Grafana.

Step 2: Install Grafana

Now we will install Grafana. *If you wish to actually test Grafana skip to Section Two below before you install.*

```
sudo yum install grafana -y
```

The next command will reload the system

```
sudo systemctl daemon-reload
```

Then we will start our server and check our service with the following two commands.

```
sudo systemctl start grafana-server
sudo systemctl status grafana-server
```

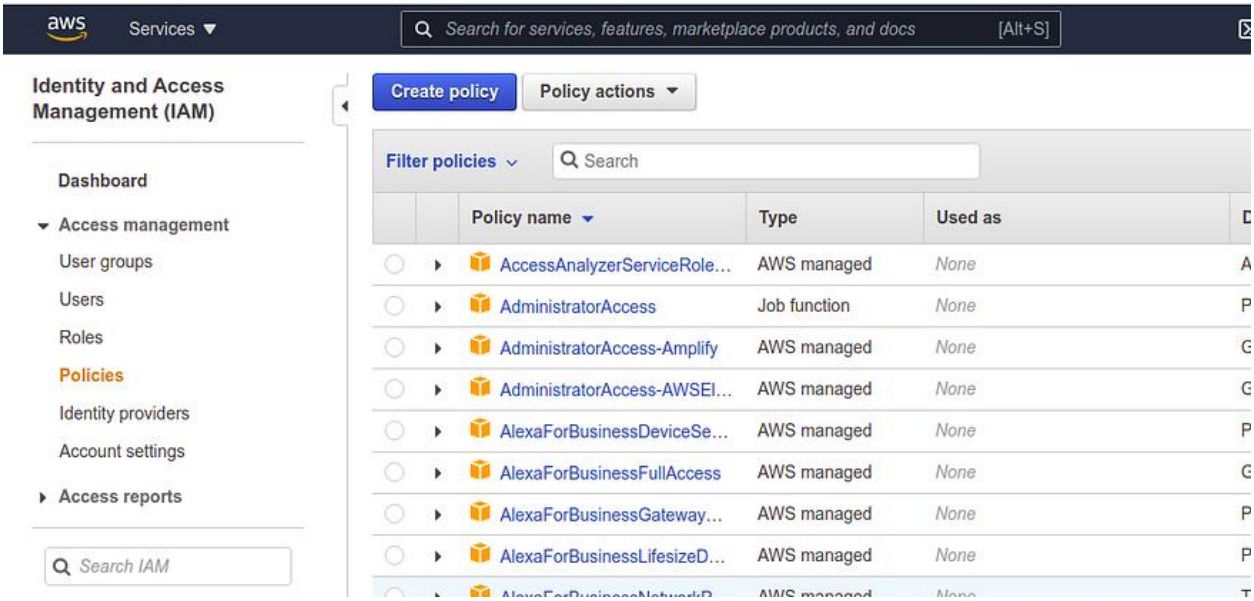
Our final command will ensure that Grafana will start up automatically if we stop and restart our instance.

```
sudo systemctl enable grafana-server.service
```

Step 3:Test Grafana

To test our server we will need to grab our public IPv4 and add a *:3000* at the end, ie. 10.90.80.10:3000, and insert it into our browser URL. This will bring you to a login screen and our Username and Password will be admin. You will be prompted to create a new password.

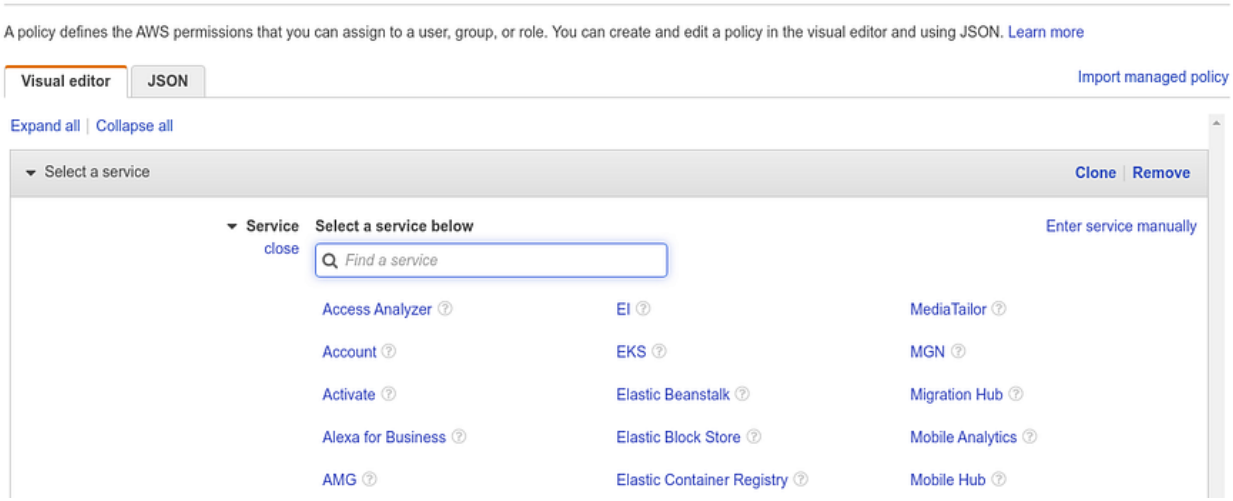
Step 4: Create a new Policy



create_policy_IAM

Just follow this, under **Services → select IAM → Policies →**

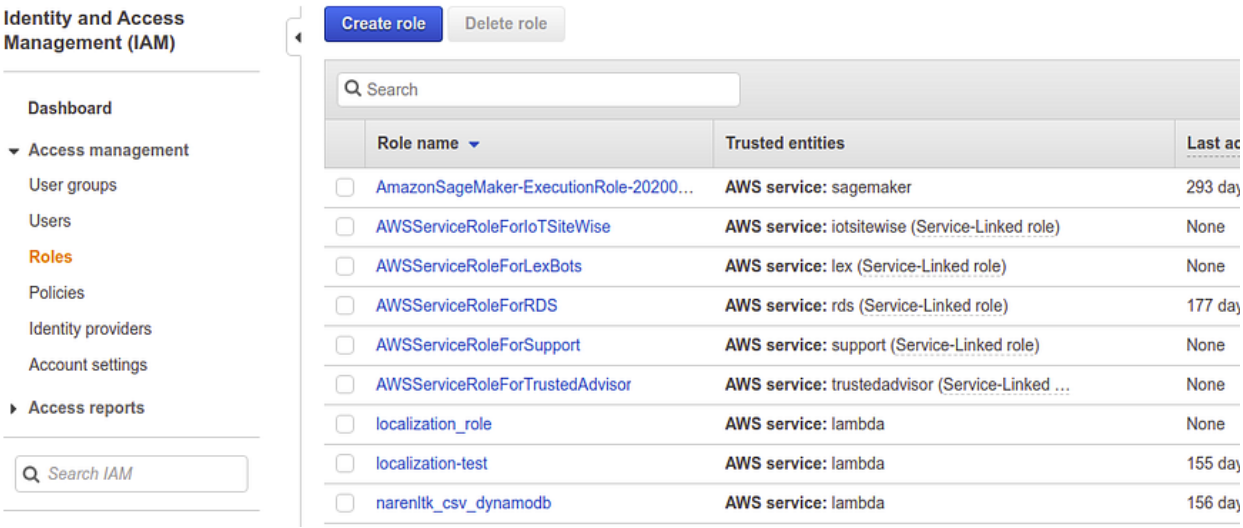
Create Policy



choose_service

Now under the services → Cloud Watch

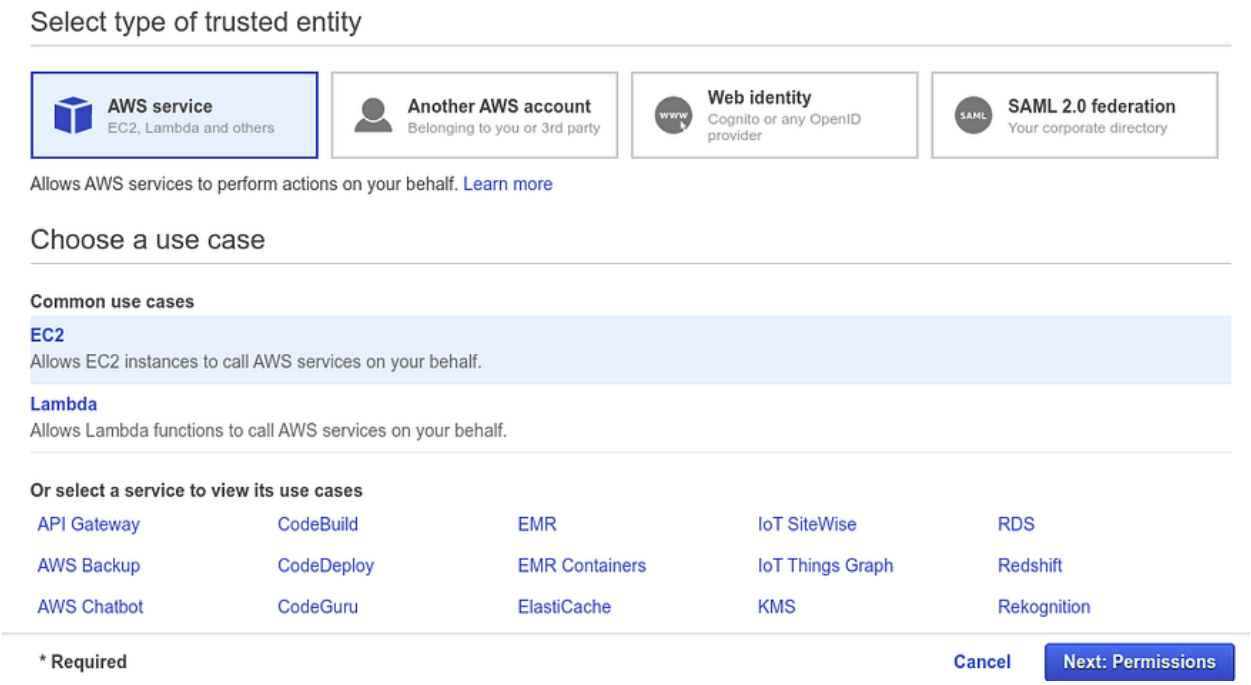
Step 5: Creating a Role



create_roles

Just follow this, under **Services → select IAM → Roles → Create Role**

Now select AWS Service → EC2 (under common use case) → next: Permission



choose_use_cases

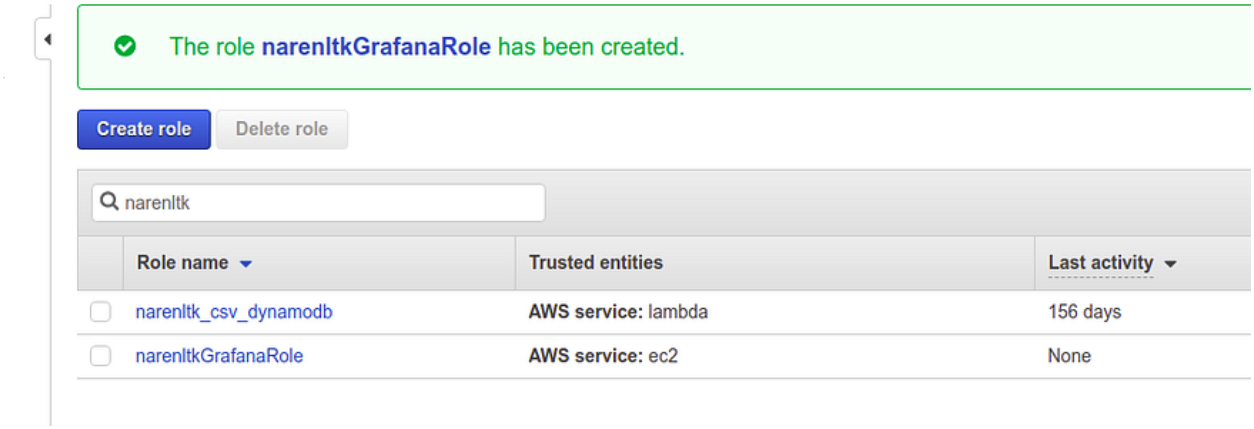
Alright now we have to attach the policy which we created earlier with this Role, as shown below,

attach_permission

Once the Policy has been attached we gotta review it, where ought to update the Role Name and Description etc.. and once we are done with this we can select Create Role.

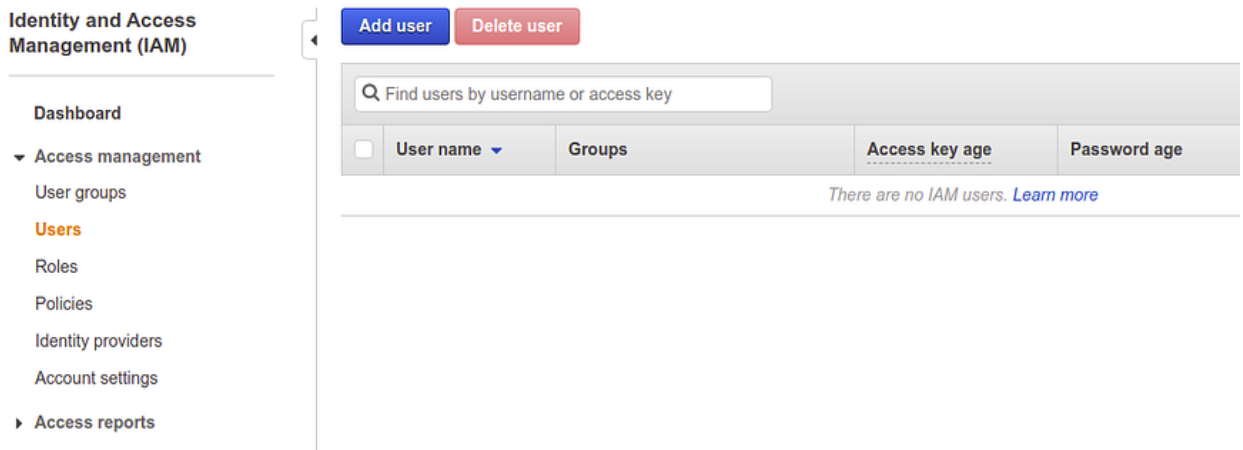
review_role

Once that is done we need to verify whether the role which we created is there or not,



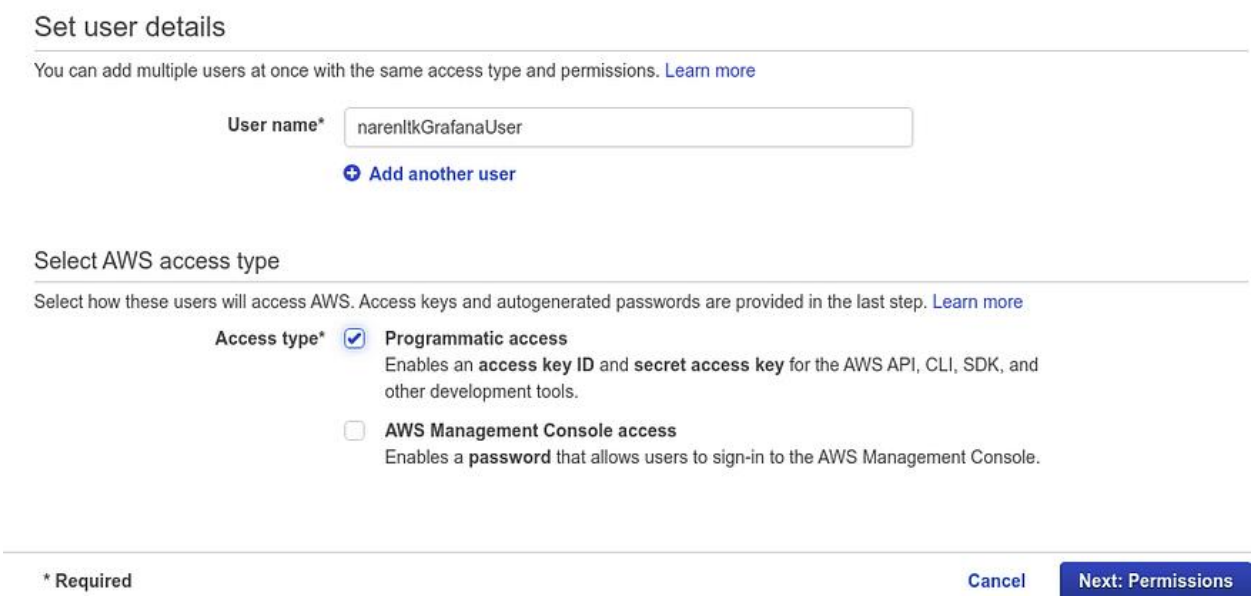
Step 6: Creating a User

First, we have created the Policy then Role and then attached the Policy with the Role and now we are creating the User so that we can attach it to the EC2 Instance.



create_iam_users

Now select the Add User.

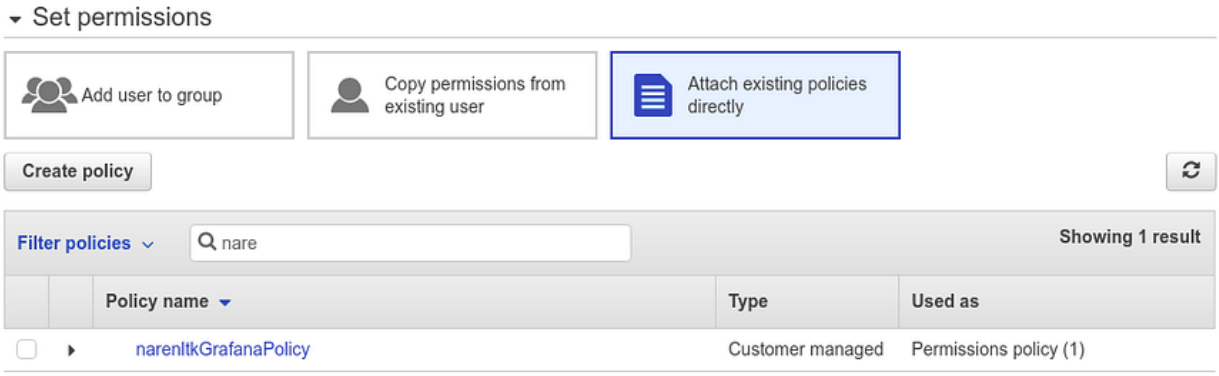


update_user_details

Give the user name.

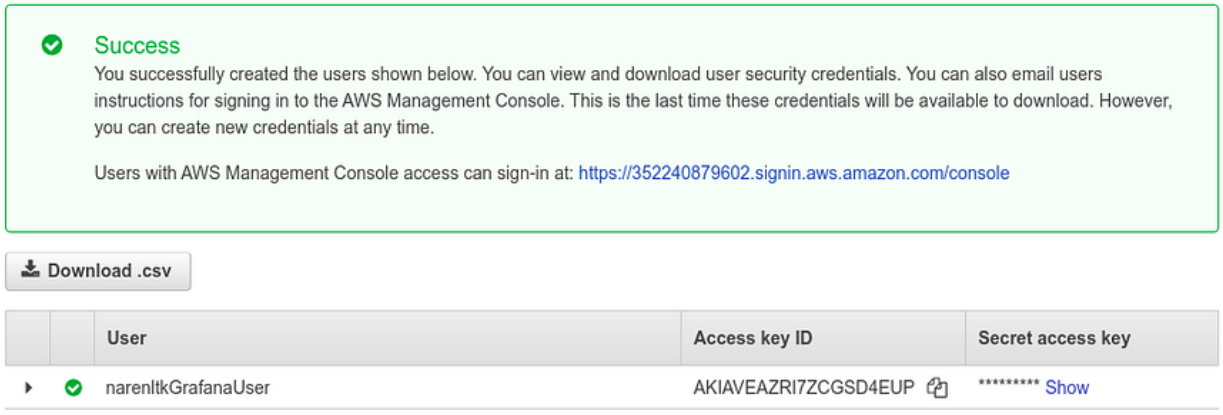
Kindly make sure that you select the programmatic access in the Access Types under select AWS access type. Which will give you the details of the Access Key and Secret Key. Make sure that you make a note of it as it will be generated only once and if you forget it then you need to create it again.

Then select Next Permission.



attach_grafana_policy

Now Set Permission, if you have already created any group means select add user to group or else proceed with Attach existing policies directly → search for the Policy which we created in the last step and proceed.



grafana_user_Cred

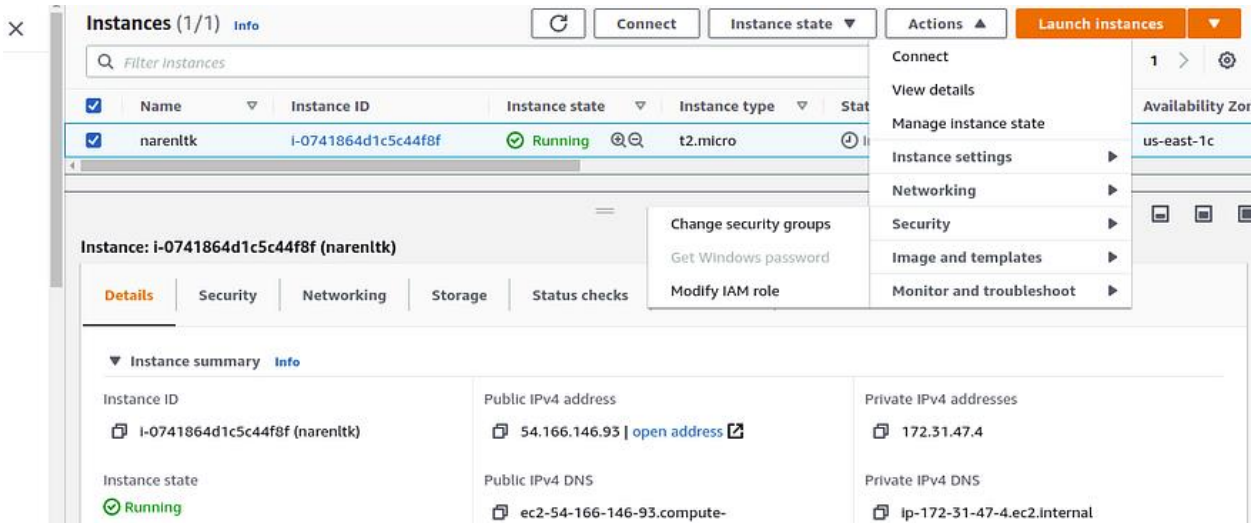
Now here you can see the Access Key ID and Secret access key, which I was talking about earlier.

Yeah, you have to save it since it will be displayed only once just like the proverb "**The Golden Words are not repeated**".

I guess now we are all set to go…. Now all that's left is just attach the User with the EC2 instance which we have.

Step 7: Attach the User to EC2 Instance

Here I do expect you all to have your EC2 instance ready i.e. it should have been started.



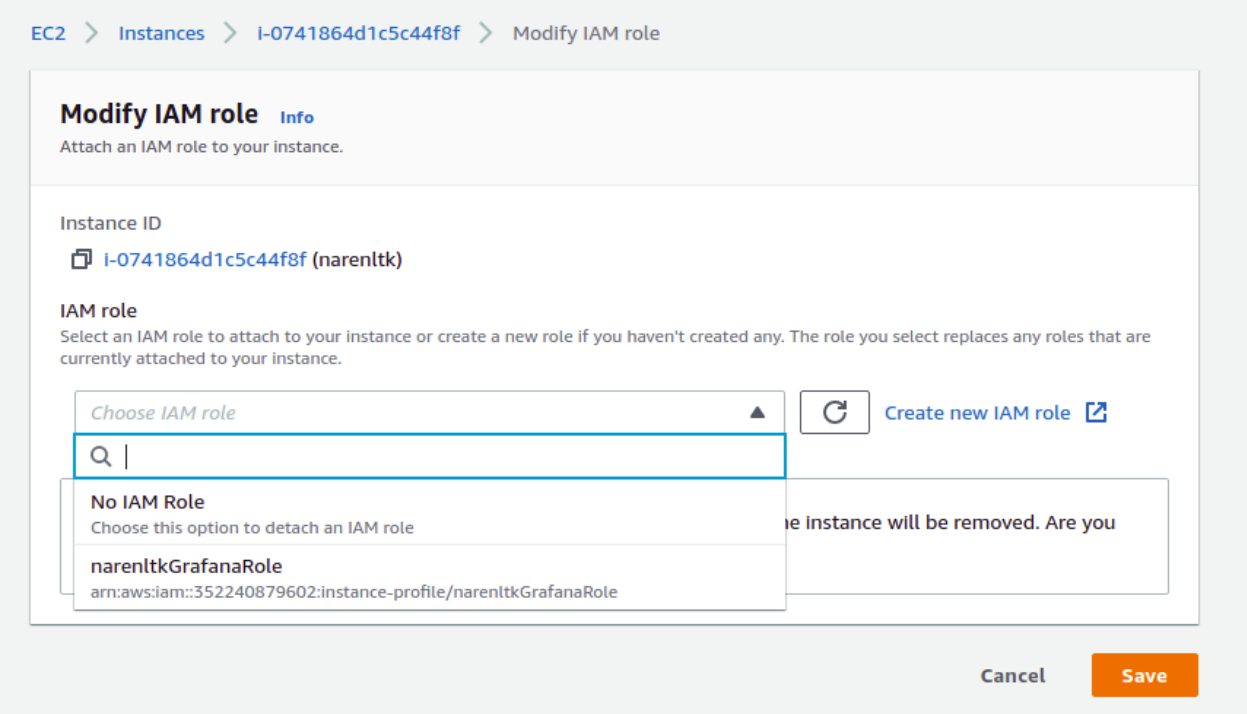
attach_iam_role

Now select the instance for which you wanted to update the IAM User and click on

Action → Security → Modify IAM Role. (This is the latest since AWS has been keep on updating)

If you have been referring to the Old videos on YouTube or else some old Medium Blogs then you would have been navigated to

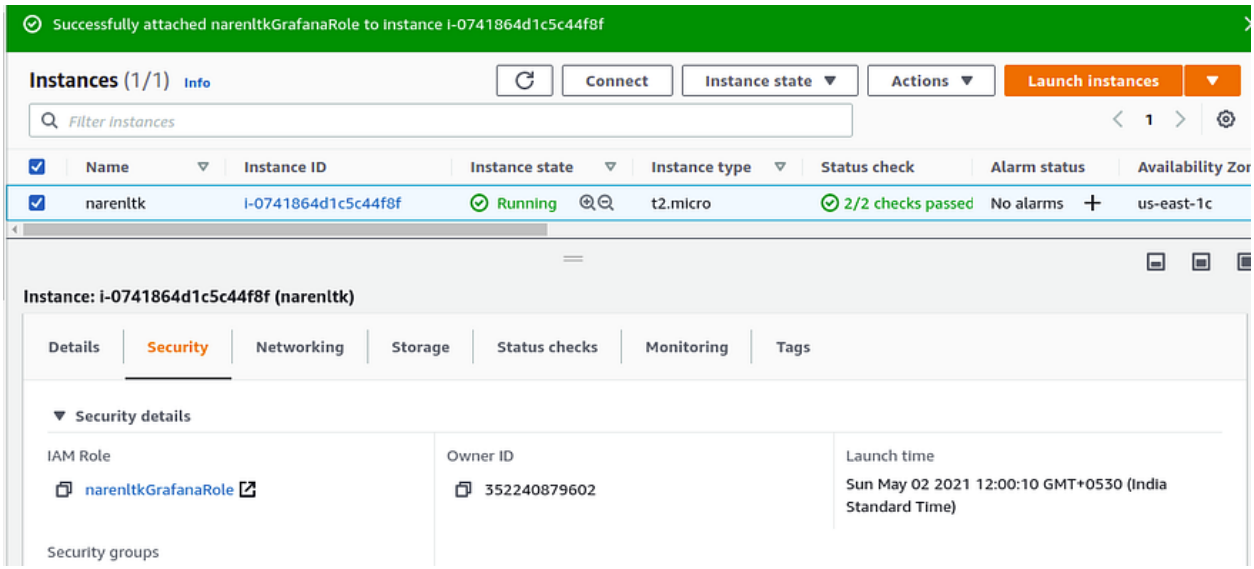
Action → Instance Settings → Attach / Replace IAM Role.



select_grafana_role

Now you gotta search for the Role which you have created and select that and save it.

Now you ought to verify that too, so kindly check the security group settings of the EC2 instance as shown below,



verify_security_tab_iam_role

You can verify the IAM Role that is attached to the Security Group.

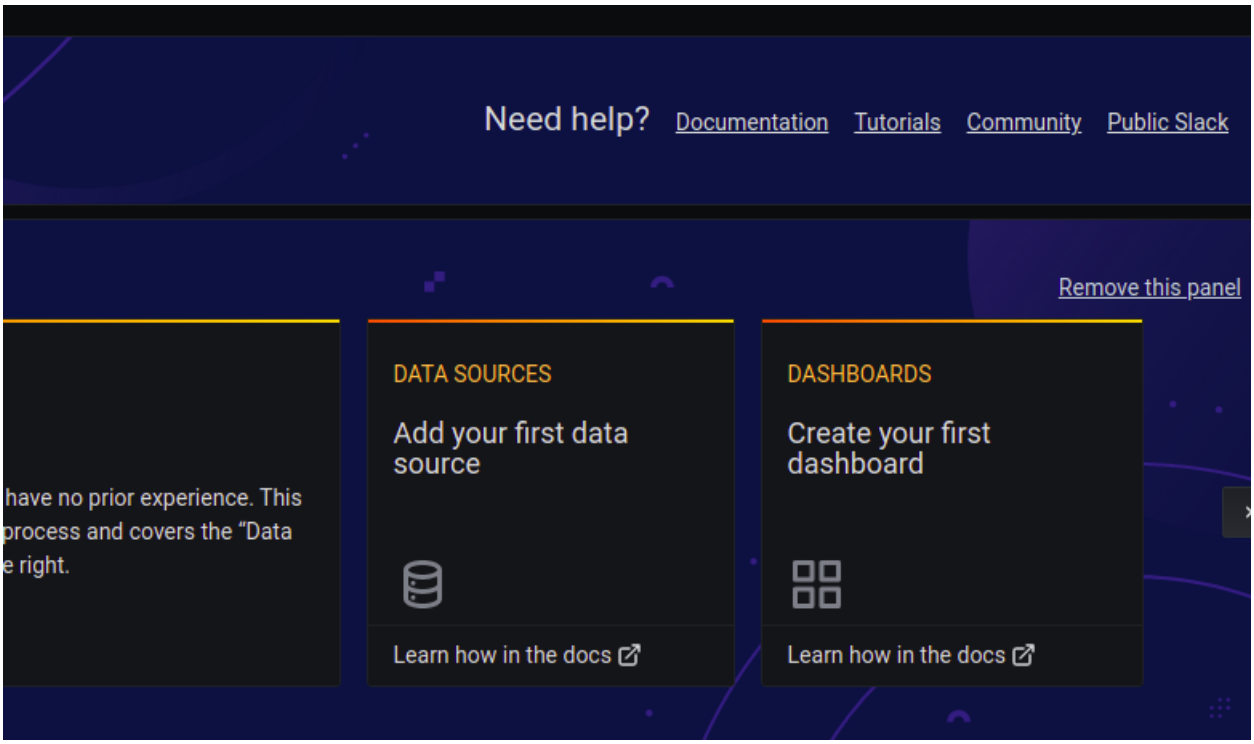
Step 8: Configure Grafana with AWS Cloud Watch

Now log into to your Grafana Dashboard, with uname and pwd which you have configured,

Note:Default uname and pwd is given below

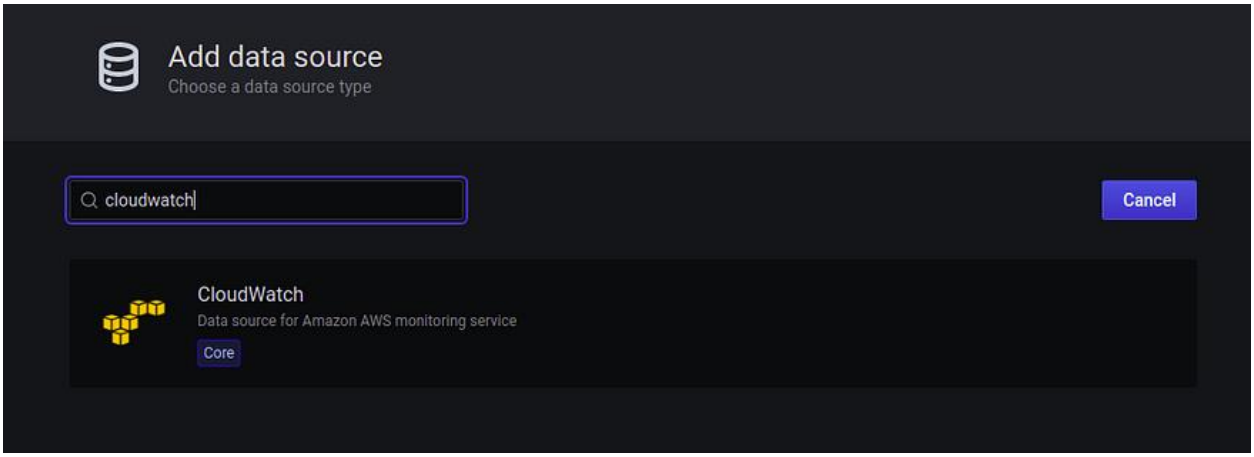
Uname → admin

Pwd → admin



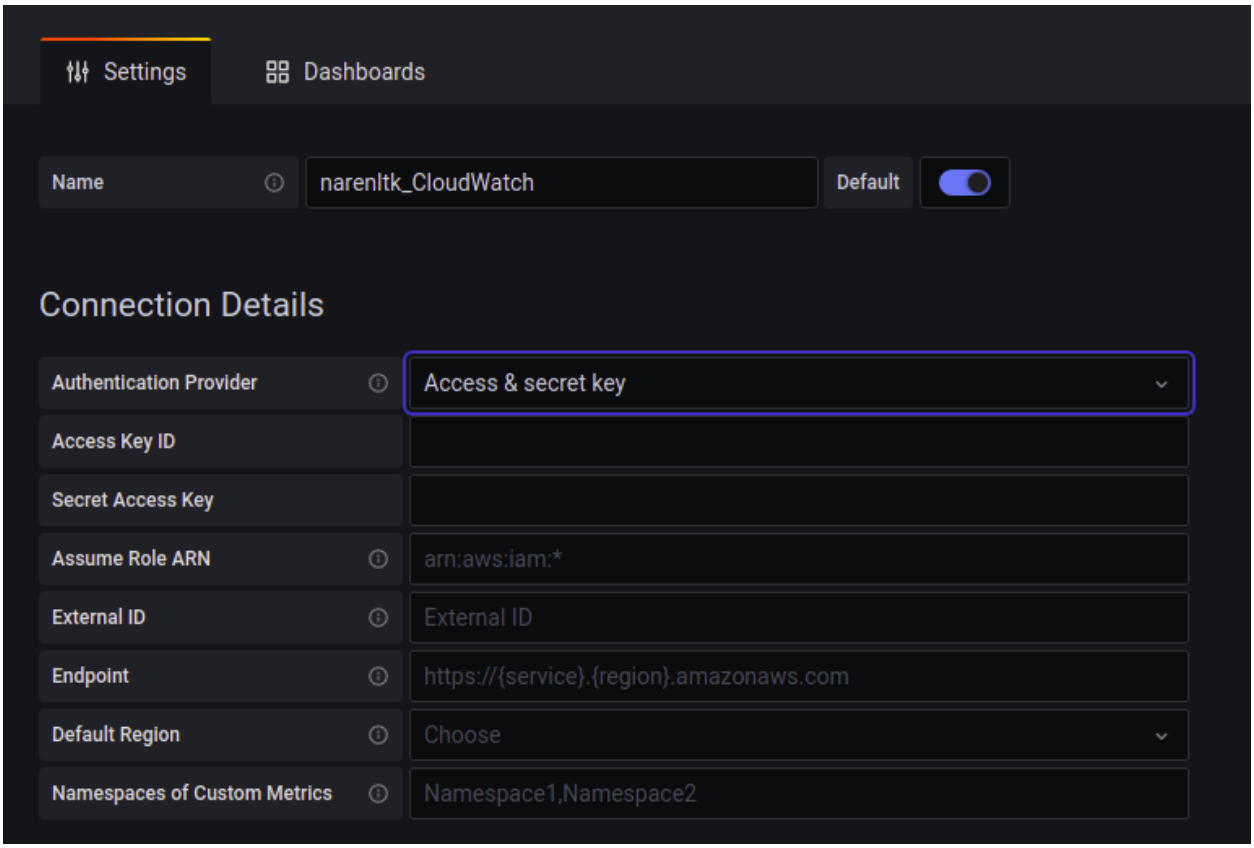
select_data_source

Here you ought to see a dashboard panel like shown above and kindly select the Data Source, i.e Add your first data source.



search_cloud_watch

Now search for the Cloud Watch in the search bar, then select it to configure the credential details.

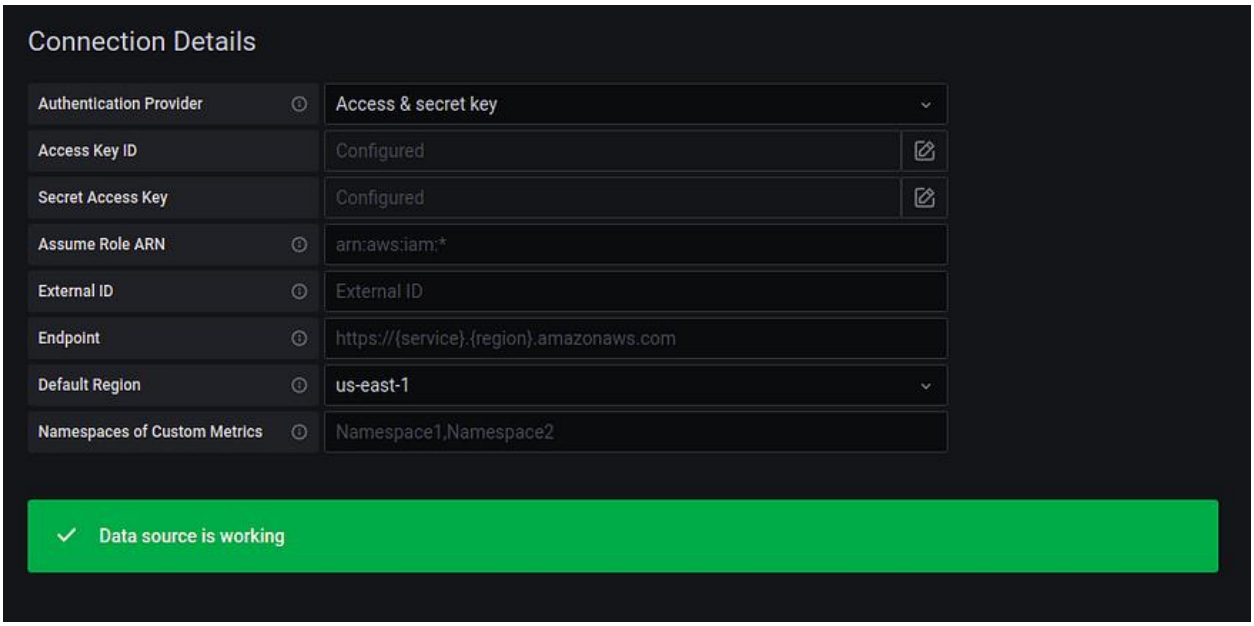


udpate_access_secret_key

Here you gotta give your desired name, then under Authentication Provider select the Access & Secret Key.

Now given the details of the Access Key and Secret Access Key alone with the Default Region.

Then click on Save and Test and you should get the following as the output.



verify_udpated_details_correct

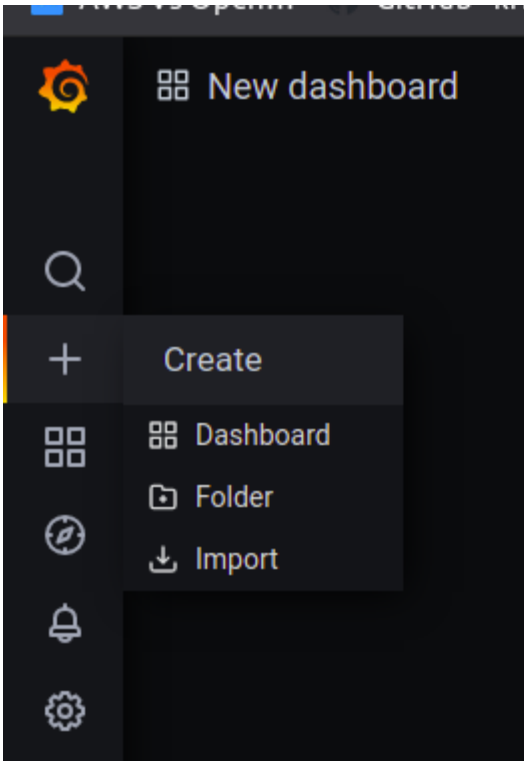
If in any place you find any error which you wanna track then, go to the EC2 instance Terminal and give the following com**mand to check the log reports,**

```
cd /var/log/grafana/lsvi grafana.log (or else) nano grafana.log
```

This will give you all the details of the error which you have faced.

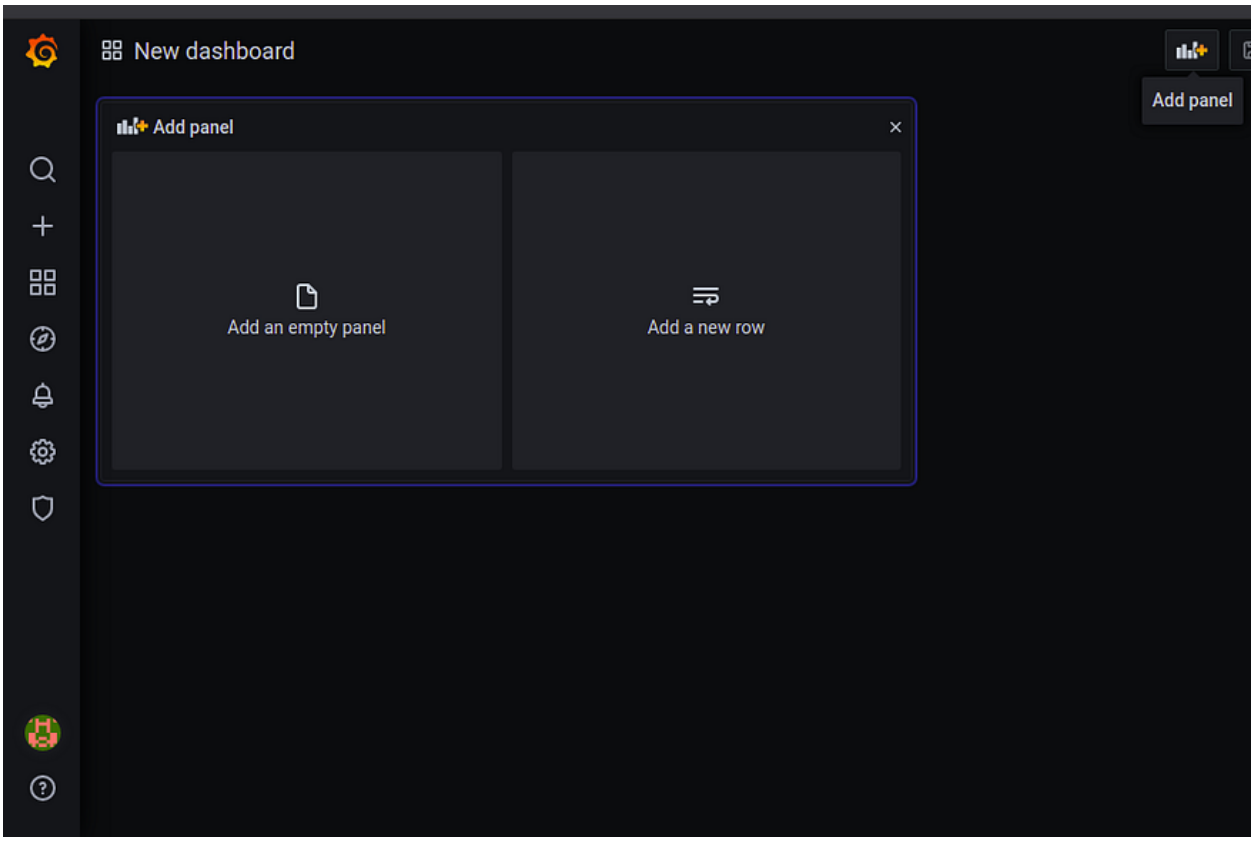
Step 9: Create your beautiful Dashboard

Now you need to select the “ + → **Dashboard** “



click_dashboard_for_making_panel

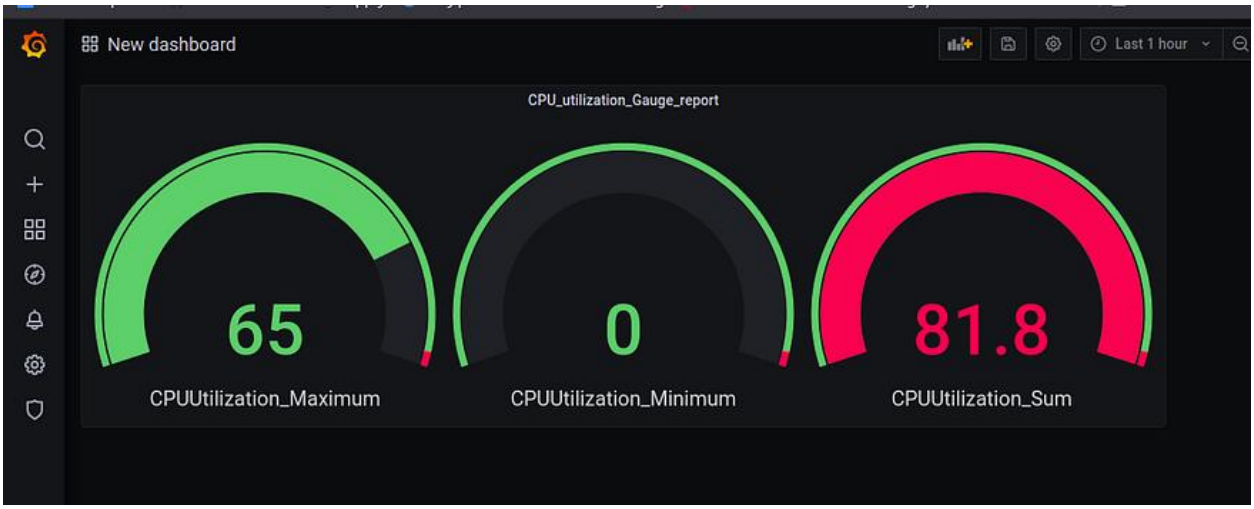
Now you can either select the Add Panel in the top right corner as sometimes panel will be created created with Add Empty Panel and Add Row as shown below,



select _add_panel

Then all the grafana query I will let you guys do it since it is easy though.

Here I have done an example with the CPU utilization.



Conclusion:

Certainly! Integrating AWS CloudWatch with Grafana enables seamless visualization and monitoring of AWS resources. Grafana's dynamic dashboard creation offers flexibility in analyzing various metrics from AWS services. Whether you choose to use the Amazon CloudWatch data source, AWS Observability CloudWatch metrics with the AWS Observability app, Grafana Alloy with the CloudWatch exporter, or the yet-another-cloudwatch-exporter (YACE), each option leverages Amazon CloudWatch APIs to make the data available in Grafana. Remember that Grafana Cloud provides an easy way to start observing your AWS environment. Good luck with your case study! 😊