CLOUD COMPUTING WITH AWS SERVICES PROJECT

Submitted by:

Himanshi Gupta

Project 1

Serverless Image Processing

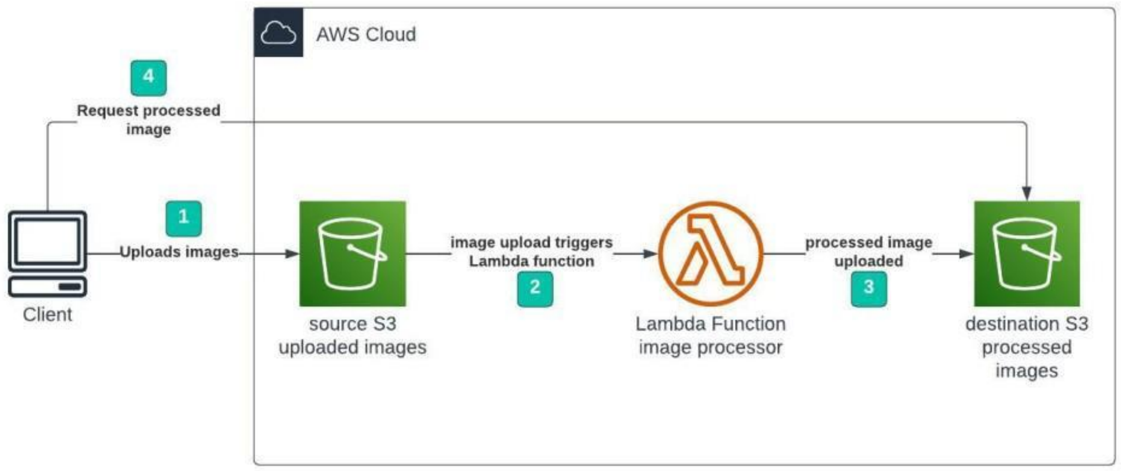
Serverless Image Processing Flow

1. The user uploads a file to the source S3 bucket (which is used for storing uploaded images).

2. When the image is uploaded to a source S3 bucket, it triggers an event that invokes the Lambda function. The lambda function processes the image.

3. The processed image is stored in the destination S3 bucket.

4. The processed image is requested by the user.


AWS Cloud
4
Request processed image
1
Uploads images
Client
source S3 uploaded images
image upload triggers Lambda function
2
Lambda Function image processor
processed image uploaded
3
destination S3 processed images

Serverless Image Processor

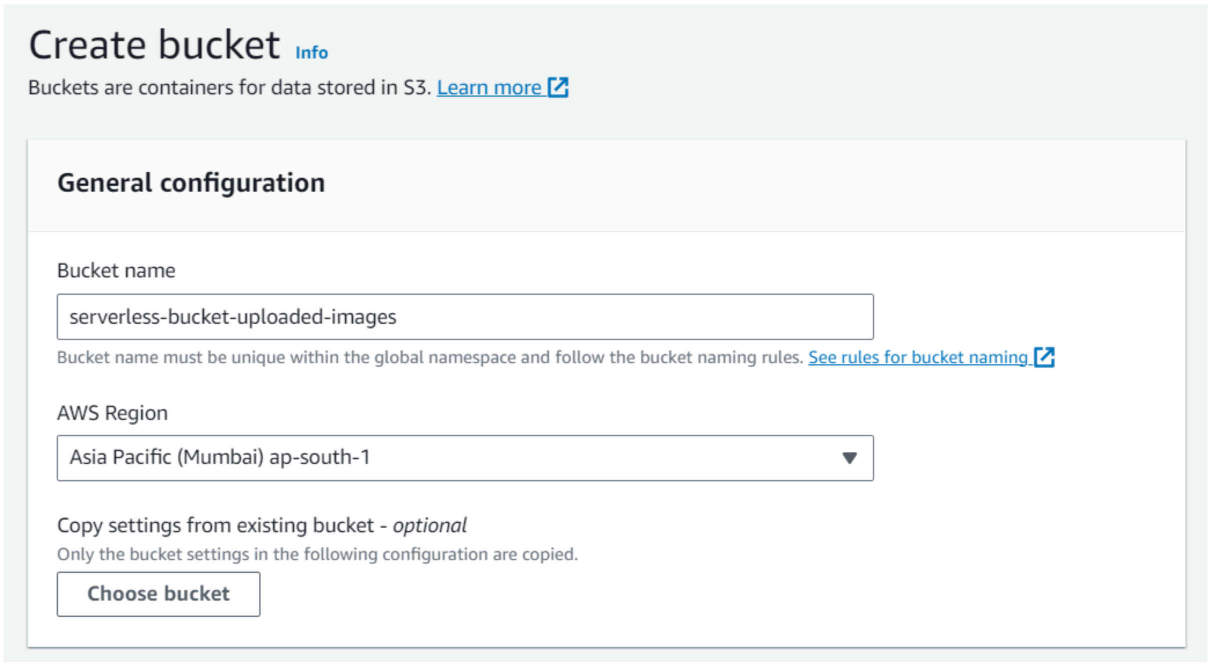
Step 1 – Creating S3 buckets

We will use two S3 buckets:

1. **source Bucket:** For storing uploaded images.

2. **destination Bucket:** For storing processed images.

Go to the S3 console and click Create bucket. Enter bucket name as 'serverless-bucket-uploaded-images'. Choose any AWS region as 'ap-south-1'.



Create bucket Info

Buckets are containers for data stored in S3. Learn more

General configuration

Bucket name

serverless-bucket-uploaded-images

Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

AWS Region

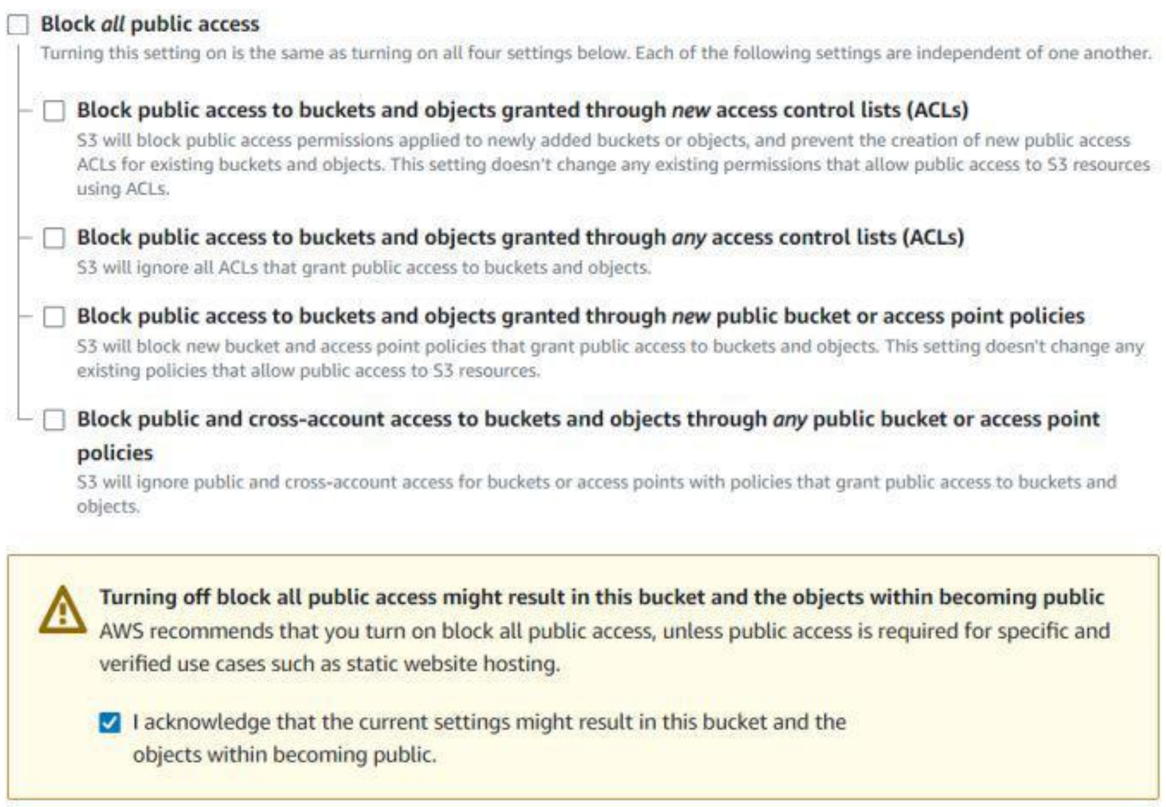
Asia Pacific (Mumbai) ap-south-1

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Step 2 – Configuring the S3 bucket policy

In the 'Block Public Access settings for this bucket' section disable "block all public access". You will get a warning that the bucket and its objects might become public. Agree to the warning**. (Note: we are making this bucket public only for this project, it is not recommended to make an S3 bucket public if not needed)**.



☐ **Block *all* public access**
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ **Block public access to buckets and objects granted through *new* access control lists (ACLs)**
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ **Block public access to buckets and objects granted through *any* access control lists (ACLs)**
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ **Block public access to buckets and objects granted through *new* public bucket or access point policies**
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ **Block public and cross-account access to buckets and objects through *any* public bucket or access point policies**
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

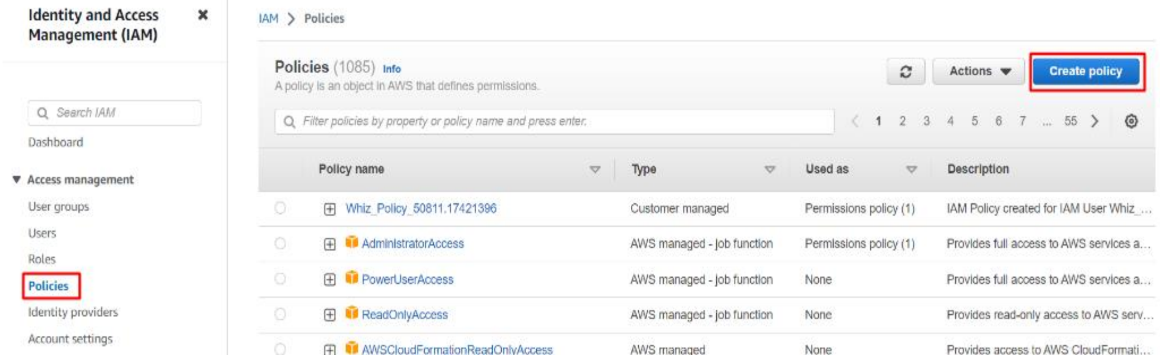
Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☑ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

Leave all other settings as default and create a bucket. Similarly, create another bucket named 'serverless-bucket-processed-images' with the same region.

STEP 3: Create an IAM Policy

1. As a pre-requisite for creating the Lambda function, we need to create a user role with a custom policy.
2. Go to **Services** and Select **IAM** under **Security, Identity, and Compliance.**
3. Click on **Policies** in the left navigation bar and click on the **Create Policy** button.



Policies (1085) Info
A policy is an object in AWS that defines permissions.

Filter policies by property or policy name and press enter.

Policy name	Type	Used as	Description
Whiz_Policy_50811.17421396	Customer managed	Permissions policy (1)	IAM Policy created for IAM User Whiz_...
AdministratorAccess	AWS managed - job function	Permissions policy (1)	Provides full access to AWS services a...
PowerUserAccess	AWS managed - job function	None	Provides full access to AWS services a...
ReadOnlyAccess	AWS managed - job function	None	Provides read-only access to AWS serv...
AWSCloudFormationReadOnlyAccess	AWS managed	None	Provides access to AWS CloudFormati...

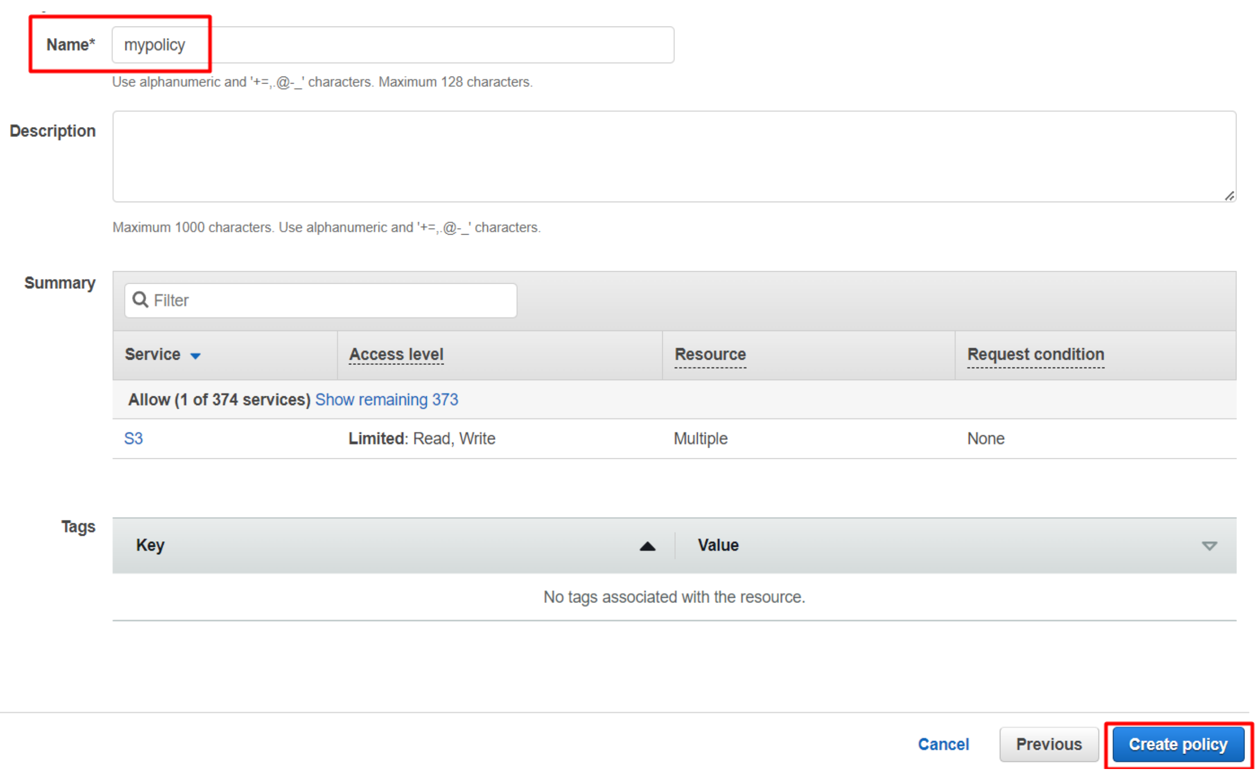
4. Click on the **JSON** tab, Remove the existing code, and copy-paste the below policy statement into the editor:

- Policy JSON:

```
{
    "Version":"2012-10-17",
    "Statement":[
       {
          "Effect":"Allow",
          "Action":[
             "s3:GetObject"
          ],
          "Resource":[
             "arn:aws:s3:::mysourcebucket12345/*"
          ]
       },
       {
          "Effect":"Allow",
          "Action":[
             "s3:PutObject"
          ],
          "Resource":[
             "arn:aws:s3:::mydestinationbucket12345/*"
          ]
       }
    ]
    }
```

the **Source** and **destination ARN name** of the bucket (which you have saved before) in the option **Resource**. Make sure to add **/*** at the end of the **ARN name.**

- Leave everything as default and click on the **Next** button.
- On the Review Policy Page:
- Policy Name: Enter ***mypolicy***
- Click on the **Create policy** button.



Name* mypolicy

Use alphanumeric and '+=,.@-_' characters. Maximum 128 characters.

Description

Maximum 1000 characters. Use alphanumeric and '+=,.@-_' characters.

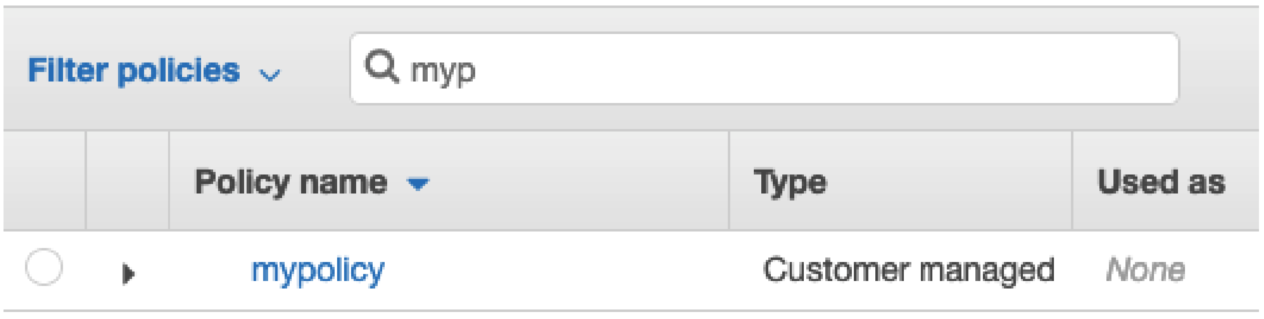
Summary

Service	Access level	Resource	Request condition
Allow (1 of 374 services) Show remaining 373			
S3	Limited: Read, Write	Multiple	None

Tags

Key	Value
No tags associated with the resource.	

5. An IAM Policy with the name **mypolicy** is created.



Policy name	Type	Used as
mypolicy	Customer managed	None

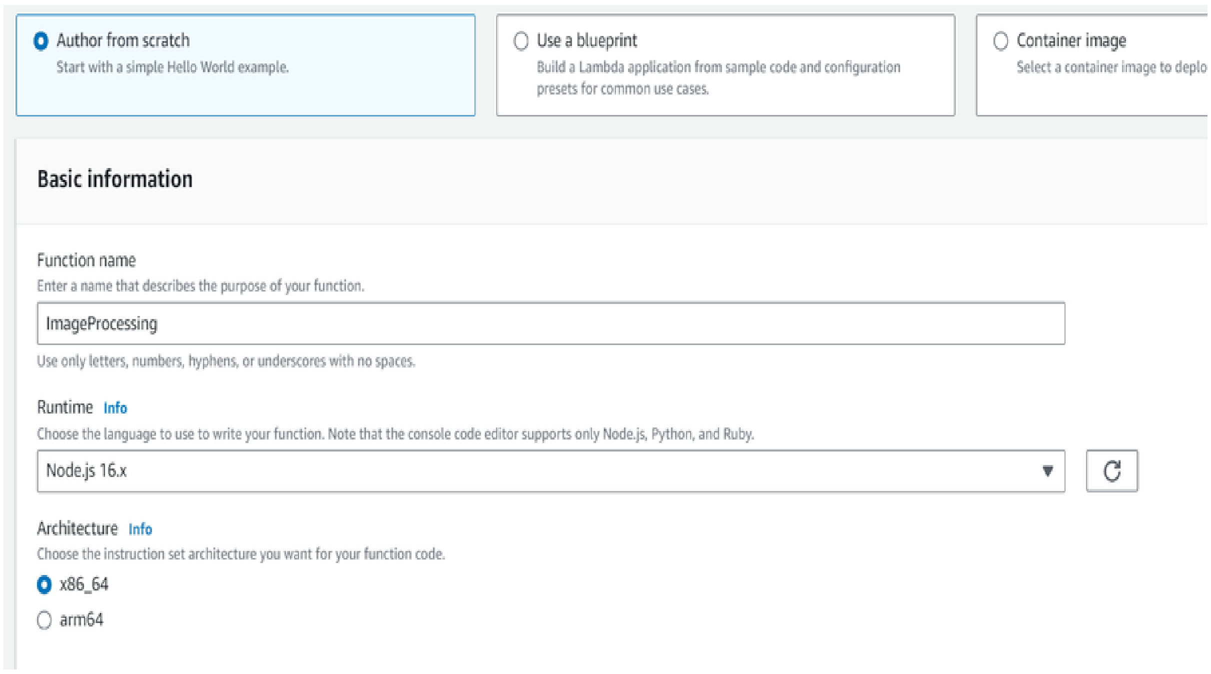
STEP 4: Create an IAM Role

1. In the left menu, click on **Roles.** Click on the **Create role** button.
2. Select **Lambda** from AWS Services list.

- From **Trusted Entity Type**: Select **AWS Service**
- From **Use case**: Select **Lambda**
- Click on the **Next** button.

Step 5 – Creating Lambda function

Go to AWS Lambda console. Navigate to the Functions section. Click Create Function and name it "ImageProcessing". Select runtime as "NodeJS 16.x" and architecture as "x86_64". Leave all other settings as default. Create the function.


[✓ Author from scratch] Start with a simple Hello World example.
Use a blueprint — Build a Lambda application from sample code and configuration presets for common use cases.
Container image — Select a container image to deplo

Basic information

Function name
Enter a name that describes the purpose of your function.
ImageProcessing
Use only letters, numbers, hyphens, or underscores with no spaces.

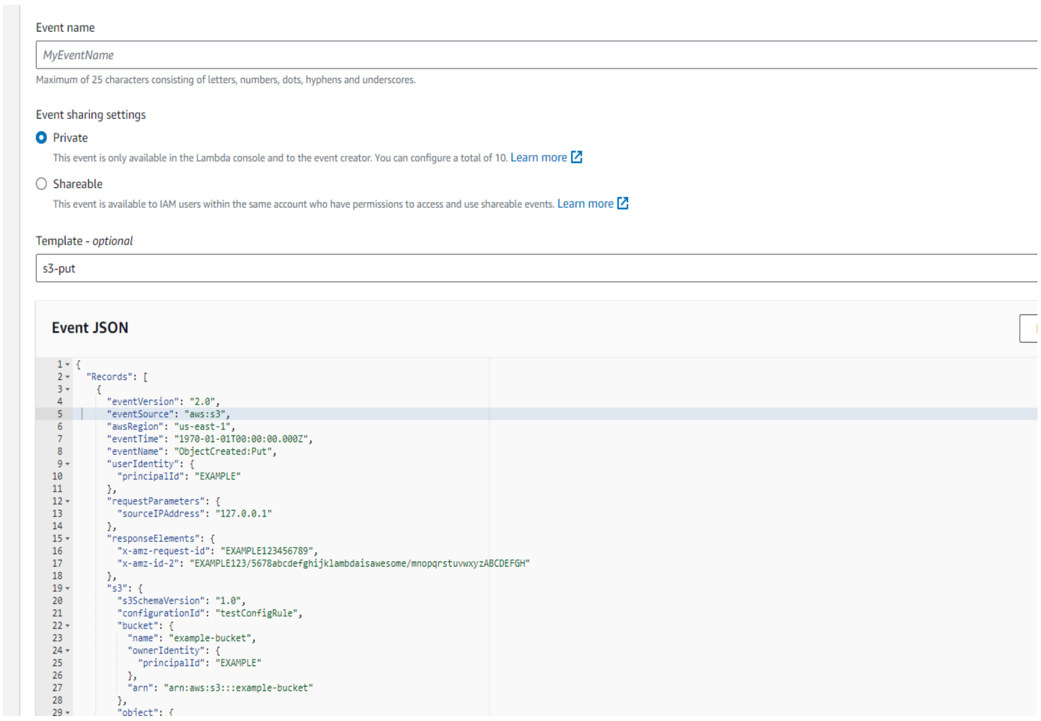
Runtime Info
Choose the language to use to write your function. Note that the console code editor supports only Node.js, Python, and Ruby.
Node.js 16.x

Architecture Info
Choose the instruction set architecture you want for your function code.
[✓ x86_64]
arm64

In the code editor on the Lambda function page upload the zip file Then click on the configuration and go to the environment variables.

- Use key: **DUST_BUCKET** and fill in the name of your destination bucket in the value block.

- Now click on the test tab and select the S3-put.


Event name

MyEventName

Maximum of 25 characters consisting of letters, numbers, dots, hyphens and underscores.

Event sharing settings

[✓ Private]

This event is only available in the Lambda console and to the event creator. You can configure a total of 10. Learn more

Shareable

This event is available to IAM users within the same account who have permissions to access and use shareable events. Learn more

Template - *optional*

s3-put

Event JSON

```
{
  "Records": [
    {
      "eventVersion": "2.0",
      "eventSource": "aws:s3",
      "awsRegion": "us-east-1",
      "eventTime": "1970-01-01T00:00:00.000Z",
      "eventName": "ObjectCreated:Put",
      "userIdentity": {
        "principalId": "EXAMPLE"
      },
      "requestParameters": {
        "sourceIPAddress": "127.0.0.1"
      },
      "responseElements": {
        "x-amz-request-id": "EXAMPLE123456789",
        "x-amz-id-2": "EXAMPLE123/5678abcdefghijklambdaisawesome/mnopqrstuvwxyzABCDEFGH"
      },
      "s3": {
        "s3SchemaVersion": "1.0",
        "configurationId": "testConfigRule",
        "bucket": {
          "name": "example-bucket",
          "ownerIdentity": {
            "principalId": "EXAMPLE"
          },
          "arn": "arn:aws:s3:::example-bucket"
        },
        "object": {
```

- Change the example bucket to the source bucket and also change "test%2Fkey", to your uploaded image name. know it's ready to see so click on the test button

STEP 6: Testing the application

Upload an image file to the source S3 bucket ("serverless-bucket-uploaded-images"). Wait for a few seconds and check the destination bucket ("serverless-bucket-processed-images"). There you will see two images
(thumbnail and cover photo).
Congratulations, you just built a serverless Image-processing application.

Project 2
Deploy a Static Website on AWS

STEP 1: Sign in to AWS Management Console

Click on the **Open Console** button, and you will get redirected to AWS Console in a new browser tab.

STEP 2: Creating a S3 Bucket

In this task, we are going to create a new S3 bucket in the US East (N. Virginia) region with a unique name disabling ACLs, and allowing public access for hosting the static website.

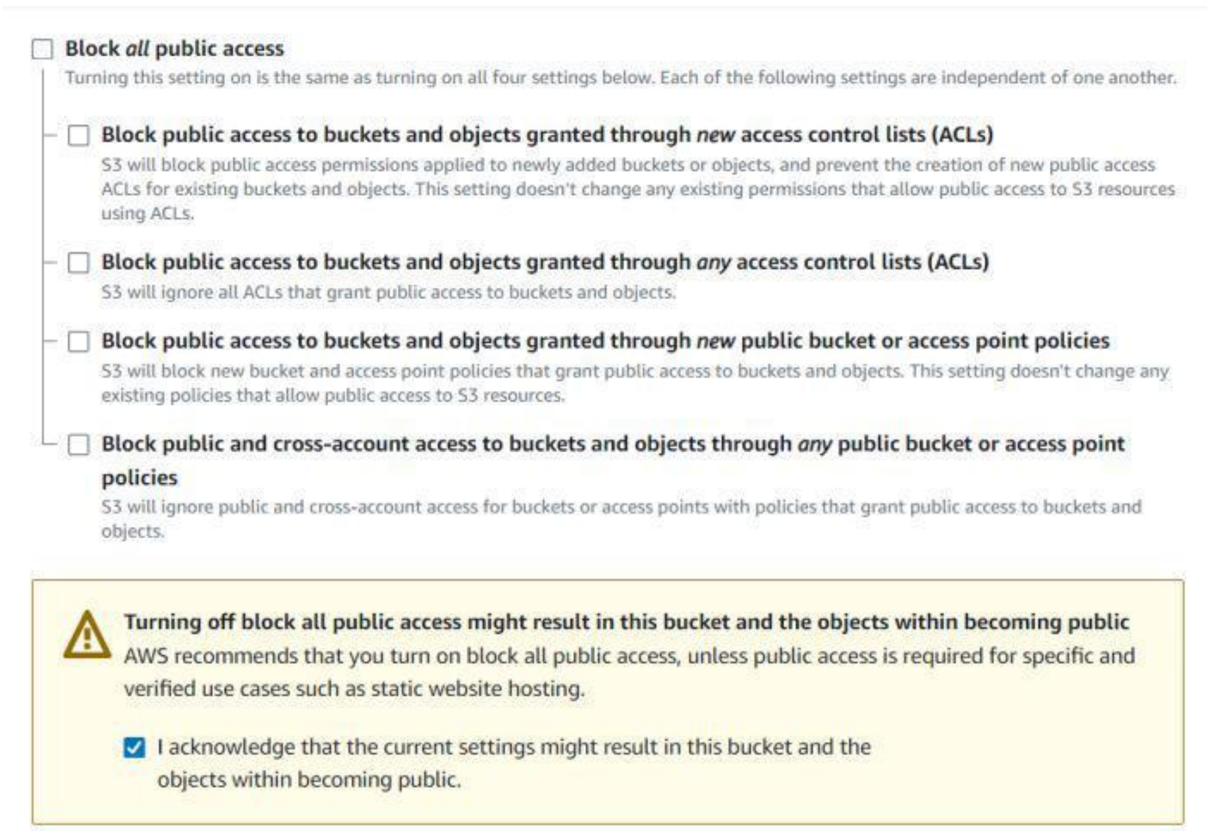
1. Navigate to **S3** by clicking on the **Services** menu at the top, then click on **S3** in the **Storage** section.
2. In the S3 dashboard, click on the **Create Bucket** button.
3. In the General Configuration, **Bucket name**: Enter **STATICWEB**

- **Note:** S3 Bucket names are globally unique, choose an available name. Maybe you can enter your name and create one.

4. AWS Region: Select **US East (N. Virginia) us-east-1**
5. Object ownership: Select the **ACLs disabled (recommended)** option
6. In the option of **Block Public Access settings for this bucket**, **Uncheck** the option of **Block all public access**.

- **Check** the I acknowledge that the current settings might result in this bucket and the objects within becoming public checkboxes.

7. Keep everything default and click on the **Create Bucket** button.


☐ Block *all* public access
Turning this setting on is the same as turning on all four settings below. Each of the following settings are independent of one another.

☐ Block public access to buckets and objects granted through *new* access control lists (ACLs)
S3 will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.

☐ Block public access to buckets and objects granted through *any* access control lists (ACLs)
S3 will ignore all ACLs that grant public access to buckets and objects.

☐ Block public access to buckets and objects granted through *new* public bucket or access point policies
S3 will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.

☐ Block public and cross-account access to buckets and objects through *any* public bucket or access point policies
S3 will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

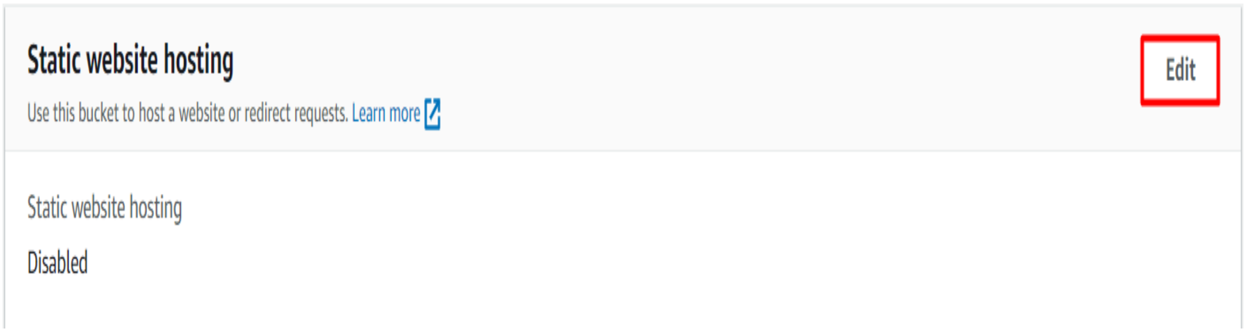
Turning off block all public access might result in this bucket and the objects within becoming public
AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting.

☑ I acknowledge that the current settings might result in this bucket and the objects within becoming public.

STEP 3: Enable Static Website Hosting settings

In this task, we will enable static website hosting for our S3 bucket, configure it to use index.html and error.html, copy the provided endpoint, upload two files, and configure the bucket policy by copying its ARN and pasting the provided policy code.

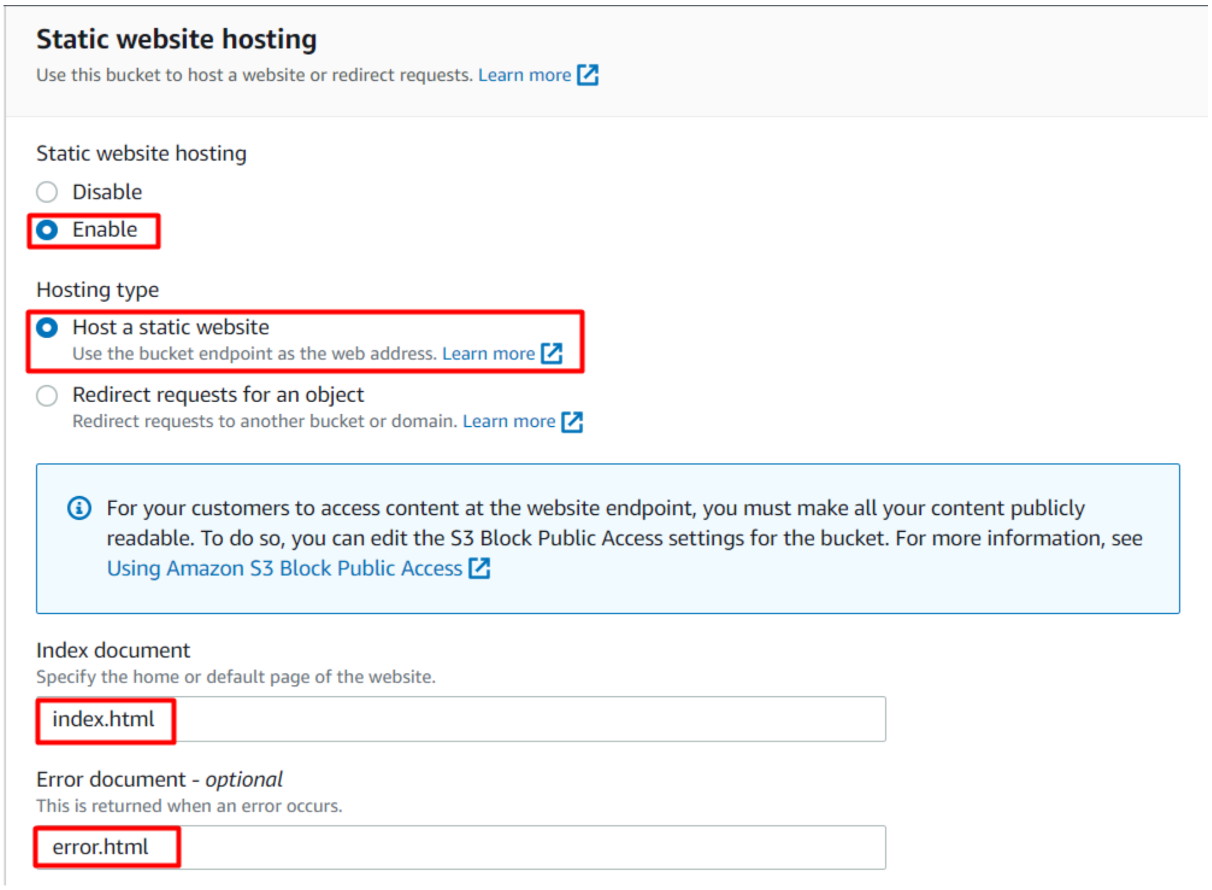
1. To proceed, go to the **S3 bucket name** that you created and click on it. After that, navigate to the **Properties** tab which can be found at the top of the screen.
2. Scroll down to the **Static website hosting** section and click on the **Edit** button.


Static website hosting
Use this bucket to host a website or redirect requests. Learn more

Static website hosting
Disabled

3. In the **Static website hosting** dialog box

- Static website hosting: Select **Enable**
- Hosting type: Choose **Host a static website**
- Index document: Type ***index.html***
- Error document: Type ***error.html***
- Click on **Save Changes**.



Static website hosting

Use this bucket to host a website or redirect requests. Learn more

Static website hosting

- Disable
- [✓ Enable]

Hosting type

- [✓ Host a static website]
  Use the bucket endpoint as the web address. Learn more
- Redirect requests for an object
  Redirect requests to another bucket or domain. Learn more

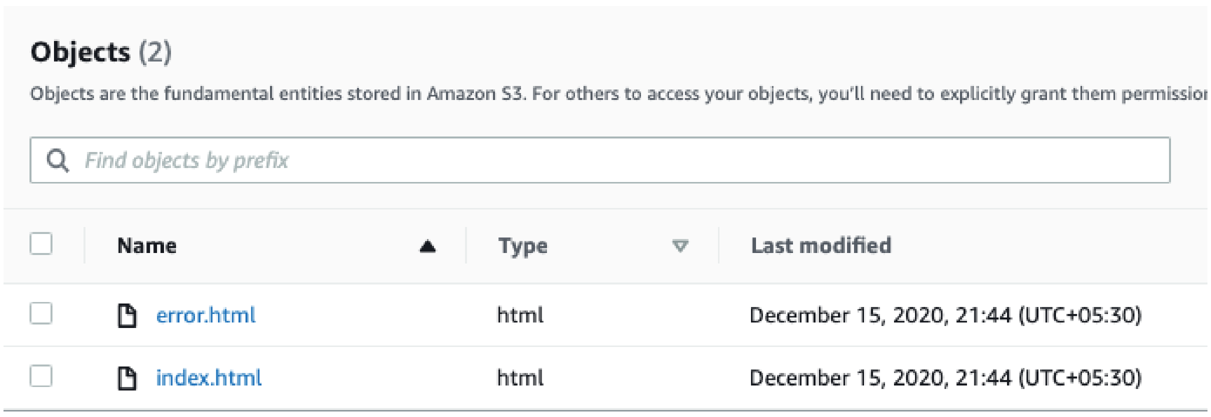
For your customers to access content at the website endpoint, you must make all your content publicly readable. To do so, you can edit the S3 Block Public Access settings for the bucket. For more information, see Using Amazon S3 Block Public Access

Index document
Specify the home or default page of the website.
index.html

Error document - *optional*
This is returned when an error occurs.
error.html

4. Go to the **Properties** tab of your S3 bucket, and find the **Static website hosting** section. **Copy** the Endpoint provided in this section to your clipboard and **save** it for future reference.

5. The next step is to download the zip file by clicking on the link, extract it, and upload two files named **index.html** and **error.html** to the S3 bucket you created earlier.


Objects (2)

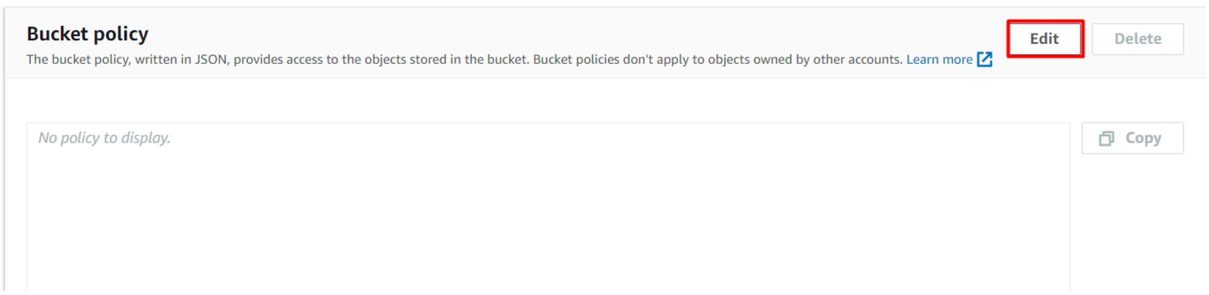
Objects are the fundamental entities stored in Amazon S3. For others to access your objects, you'll need to explicitly grant them permissio

Find objects by prefix

Name	Type	Last modified
error.html	html	December 15, 2020, 21:44 (UTC+05:30)
index.html	html	December 15, 2020, 21:44 (UTC+05:30)

6. To configure your S3 bucket, access the **Permissions** tab and make the necessary configurations.

- In the **Permissions** tab, Click on the **Edit** button beside the **Bucket Policy**.


Bucket policy

The bucket policy, written in JSON, provides access to the objects stored in the bucket. Bucket policies don't apply to objects owned by other accounts. Learn more

No policy to display.

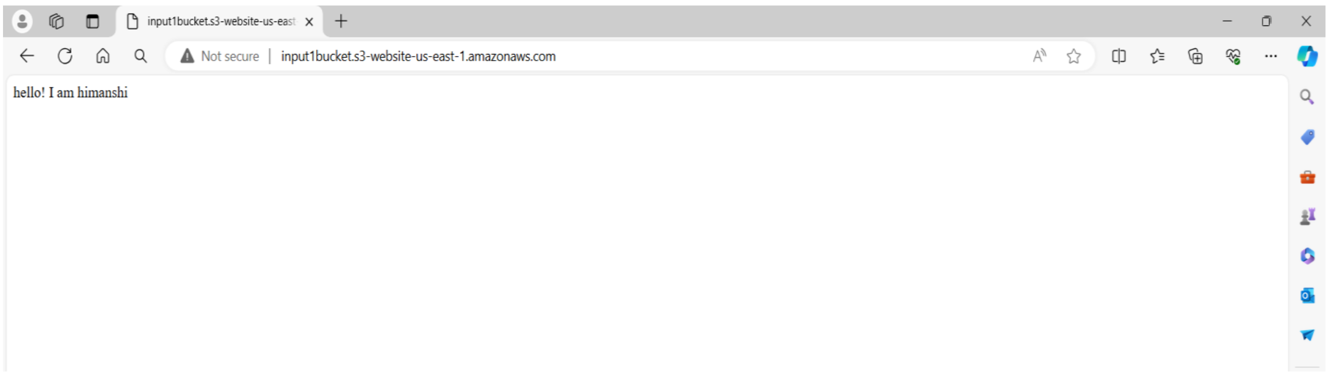
- You will be able to see a Blank policy editor.
- Before creating the policy, you will need to copy the **ARN** (Amazon Resource Name) of your bucket.
- Copy the **ARN** of your bucket to the clipboard. It is displayed at the top of the policy editor. it will look like **ARN: "arn:aws:s3:::your-bucket-name**".
- In the policy below, **Update** the bucket ARN on the Resource key value and **paste** the below policy code in the editor.

```
{
"Id":"Policy1",
"Version":"2012-10-17",
"Statement":[
{
"Sid":"Stmt1",
"Action":[
"s3:GetObject"
],
"Effect":"Allow",
"Resource":"replace-this-string-with-your-bucket-arn/*",
"Principal":"*"
}
]
}
```

Click on the **Save Changes** button.

STEP 4: Test the website

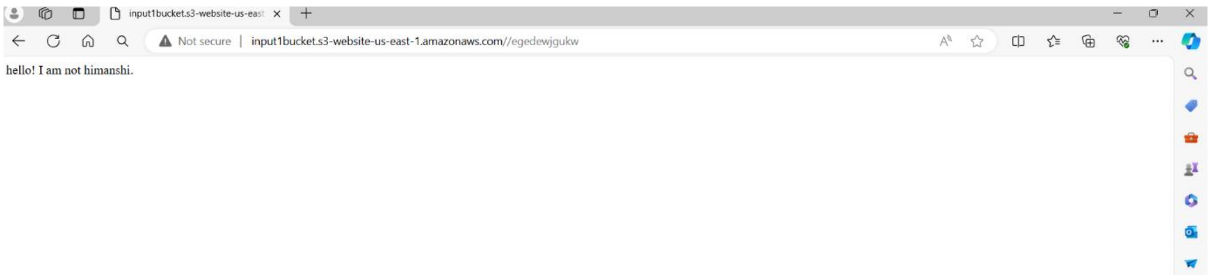
1. Now copy the **static website URL** (that we saved earlier) and run it in your browser. You will be able to see the index.html file's text. A sample screenshot is attached below:


input1bucket.s3-website-us-east
Not secure
input1bucket.s3-website-us-east-1.amazonaws.com

hello! I am himanshi

STEP 5: Test the website's error page

Copy the static website URL (which we saved earlier), but this time, add some random characters to the end of the URL to break it. When satisfied, hit **[Enter]**. You will be redirected to the **error.html** page.


input1bucket.s3-website-us-east
Not secure
input1bucket.s3-website-us-east-1.amazonaws.com//egedewjgukw

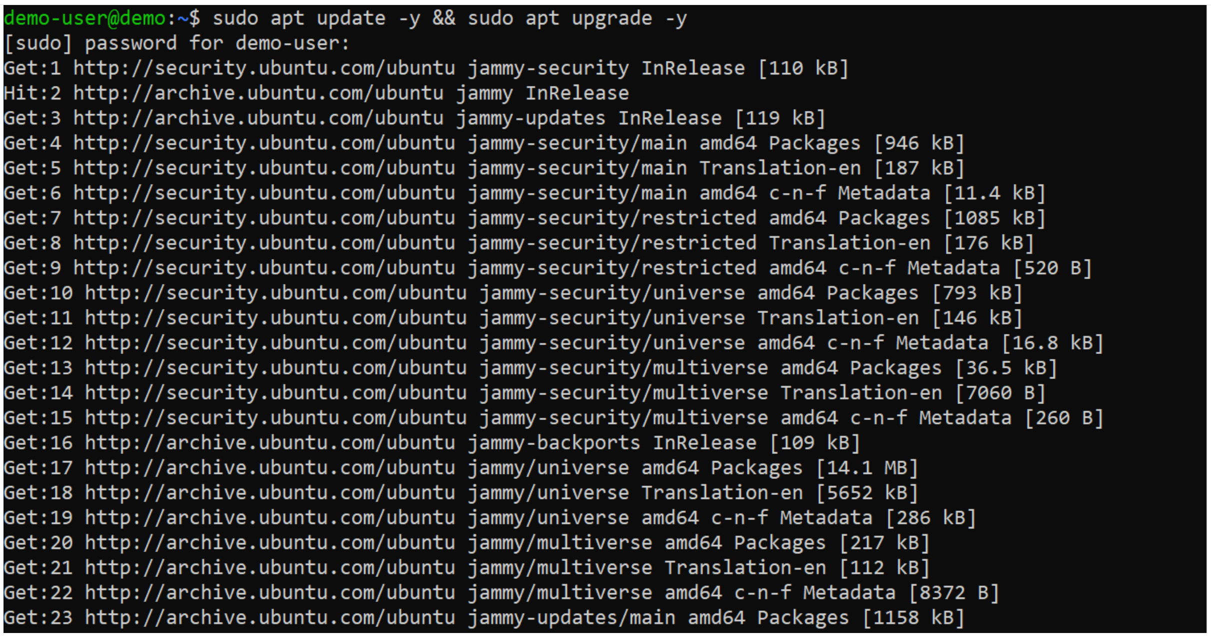
hello! I am not himanshi.

Project 3

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

Step 1 - Update and upgrade

```
sudo apt update -y && sudo apt upgrade -y
```

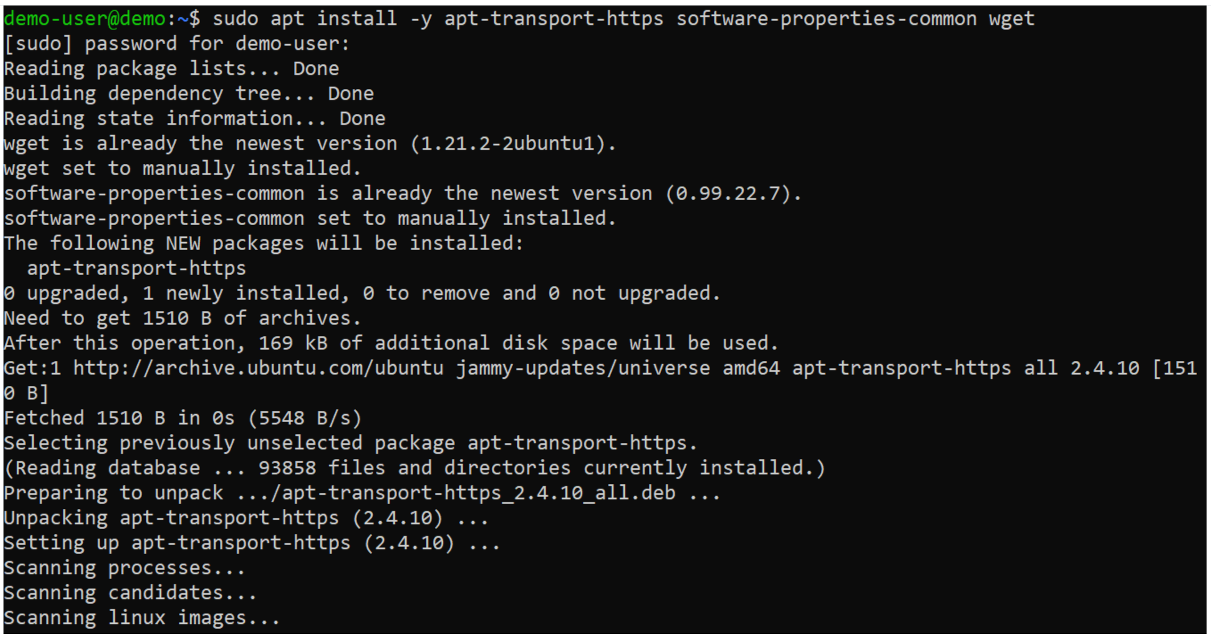


```
demo-user@demo:~$ sudo apt update -y && sudo apt upgrade -y
[sudo] password for demo-user:
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Hit:2 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:3 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Get:4 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [946 kB]
Get:5 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [187 kB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [11.4 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [1085 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/restricted Translation-en [176 kB]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 c-n-f Metadata [520 B]
Get:10 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 Packages [793 kB]
Get:11 http://security.ubuntu.com/ubuntu jammy-security/universe Translation-en [146 kB]
Get:12 http://security.ubuntu.com/ubuntu jammy-security/universe amd64 c-n-f Metadata [16.8 kB]
Get:13 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 Packages [36.5 kB]
Get:14 http://security.ubuntu.com/ubuntu jammy-security/multiverse Translation-en [7060 B]
Get:15 http://security.ubuntu.com/ubuntu jammy-security/multiverse amd64 c-n-f Metadata [260 B]
Get:16 http://archive.ubuntu.com/ubuntu jammy-backports InRelease [109 kB]
Get:17 http://archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:18 http://archive.ubuntu.com/ubuntu jammy/universe Translation-en [5652 kB]
Get:19 http://archive.ubuntu.com/ubuntu jammy/universe amd64 c-n-f Metadata [286 kB]
Get:20 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 Packages [217 kB]
Get:21 http://archive.ubuntu.com/ubuntu jammy/multiverse Translation-en [112 kB]
Get:22 http://archive.ubuntu.com/ubuntu jammy/multiverse amd64 c-n-f Metadata [8372 B]
Get:23 http://archive.ubuntu.com/ubuntu jammy-updates/main amd64 Packages [1158 kB]
```

#Step 2 - Install the required packages

Next, run the following command to install the packages needed for the installation:

```
sudo apt install -y apt-transport-https software-properties-common
wget
```

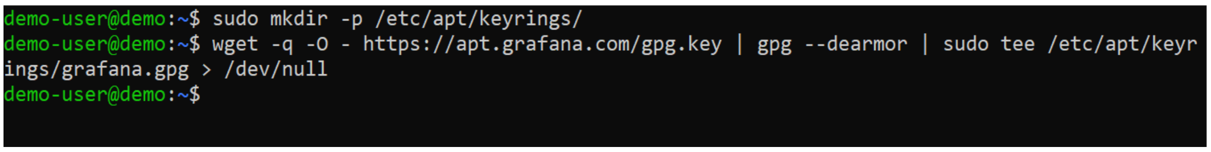


```
demo-user@demo:~$ sudo apt install -y apt-transport-https software-properties-common wget
[sudo] password for demo-user:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
wget is already the newest version (1.21.2-2ubuntu1).
wget set to manually installed.
software-properties-common is already the newest version (0.99.22.7).
software-properties-common set to manually installed.
The following NEW packages will be installed:
  apt-transport-https
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 1510 B of archives.
After this operation, 169 kB of additional disk space will be used.
Get:1 http://archive.ubuntu.com/ubuntu jammy-updates/universe amd64 apt-transport-https all 2.4.10 [151
0 B]
Fetched 1510 B in 0s (5548 B/s)
Selecting previously unselected package apt-transport-https.
(Reading database ... 93858 files and directories currently installed.)
Preparing to unpack .../apt-transport-https_2.4.10_all.deb ...
Unpacking apt-transport-https (2.4.10) ...
Setting up apt-transport-https (2.4.10) ...
Scanning processes...
Scanning candidates...
Scanning linux images...
```

#Step 3 - Add the Grafana GPG key

```
sudo mkdir -p /etc/apt/keyrings/
wget -q -O - https://apt.grafana.com/gpg.key | gpg --dearmor | sudo
tee /etc/apt/keyrings/grafana.gpg > /dev/null
```

The first command creates a directory where the key will be stored. The second command will download, convert, and store the key in the specified location for secure APT package management.



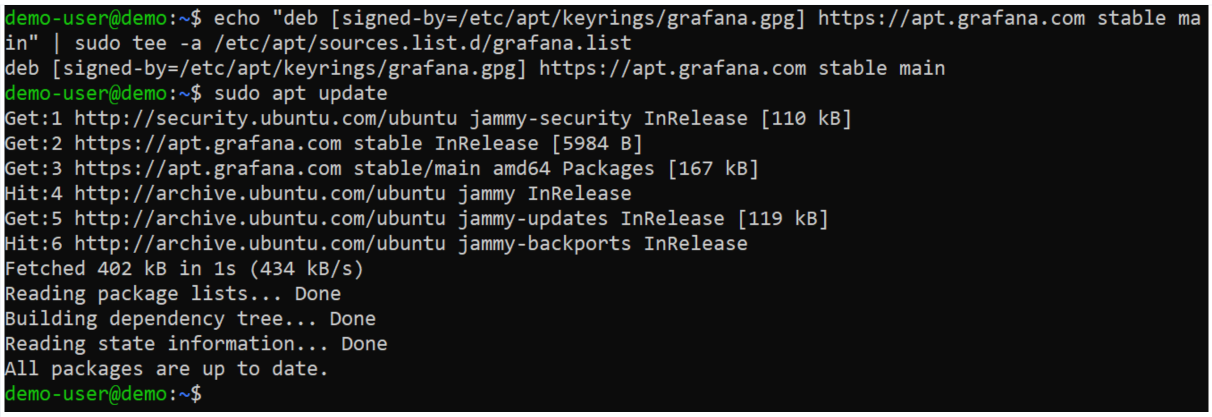
```
demo-user@demo:~$ sudo mkdir -p /etc/apt/keyrings/
demo-user@demo:~$ wget -q -O - https://apt.grafana.com/gpg.key | gpg --dearmor | sudo tee /etc/apt/keyr
ings/grafana.gpg > /dev/null
demo-user@demo:~$
```

#Step 4 - Add Grafana APT repository

```
echo "deb [signed-by=/etc/apt/keyrings/grafana.gpg]
https://apt.grafana.com stable main" | sudo tee -a
/etc/apt/sources.list.d/grafana.list
```

After adding the repository to your system, update the package index to include information from the newly added repository using:

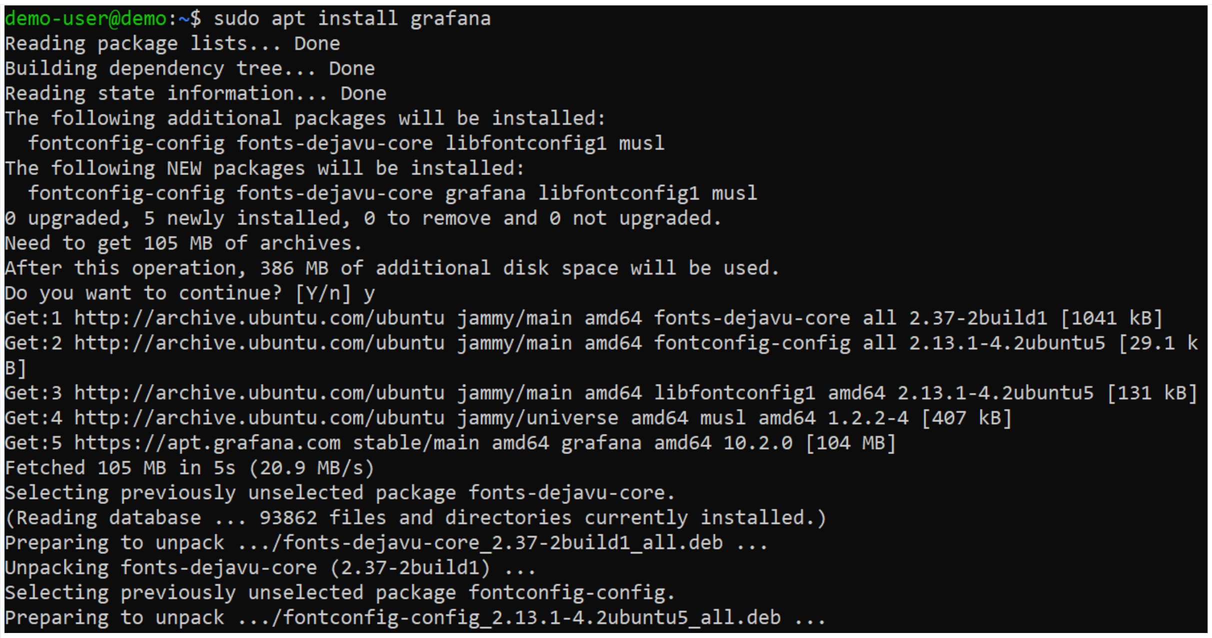
```
sudo apt update
```



```
demo-user@demo:~$ echo "deb [signed-by=/etc/apt/keyrings/grafana.gpg] https://apt.grafana.com stable ma
in" | sudo tee -a /etc/apt/sources.list.d/grafana.list
deb [signed-by=/etc/apt/keyrings/grafana.gpg] https://apt.grafana.com stable main
demo-user@demo:~$ sudo apt update
Get:1 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:2 https://apt.grafana.com stable InRelease [5984 B]
Get:3 https://apt.grafana.com stable/main amd64 Packages [167 kB]
Hit:4 http://archive.ubuntu.com/ubuntu jammy InRelease
Get:5 http://archive.ubuntu.com/ubuntu jammy-updates InRelease [119 kB]
Hit:6 http://archive.ubuntu.com/ubuntu jammy-backports InRelease
Fetched 402 kB in 1s (434 kB/s)
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
All packages are up to date.
demo-user@demo:~$
```

#Step 5 - Install Grafana

```
sudo apt install grafana
```



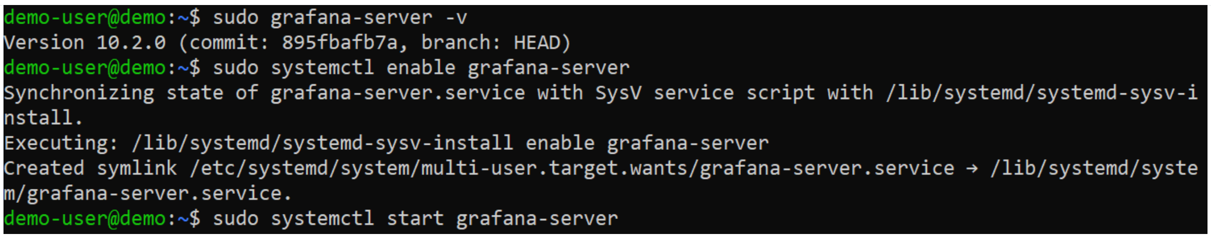
```
demo-user@demo:~$ sudo apt install grafana
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  fontconfig-config fonts-dejavu-core libfontconfig1 musl
The following NEW packages will be installed:
  fontconfig-config fonts-dejavu-core grafana libfontconfig1 musl
0 upgraded, 5 newly installed, 0 to remove and 0 not upgraded.
Need to get 105 MB of archives.
After this operation, 386 MB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://archive.ubuntu.com/ubuntu jammy/main amd64 fonts-dejavu-core all 2.37-2build1 [1041 kB]
Get:2 http://archive.ubuntu.com/ubuntu jammy/main amd64 fontconfig-config all 2.13.1-4.2ubuntu5 [29.1 k
B]
Get:3 http://archive.ubuntu.com/ubuntu jammy/main amd64 libfontconfig1 amd64 2.13.1-4.2ubuntu5 [131 kB]
Get:4 http://archive.ubuntu.com/ubuntu jammy/universe amd64 musl amd64 1.2.2-4 [407 kB]
Get:5 https://apt.grafana.com stable/main amd64 grafana amd64 10.2.0 [104 MB]
Fetched 105 MB in 5s (20.9 MB/s)
Selecting previously unselected package fonts-dejavu-core.
(Reading database ... 93862 files and directories currently installed.)
Preparing to unpack .../fonts-dejavu-core_2.37-2build1_all.deb ...
Unpacking fonts-dejavu-core (2.37-2build1) ...
Selecting previously unselected package fontconfig-config.
Preparing to unpack .../fontconfig-config_2.13.1-4.2ubuntu5_all.deb ...
```

#Step 6 - Start the Grafana service

```
sudo grafana-server -v
```

Next, start the Grafana service and enable it to start automatically at system reboot using the following commands:

```
sudo systemctl start grafana-server
sudo systemctl enable grafana-server
```

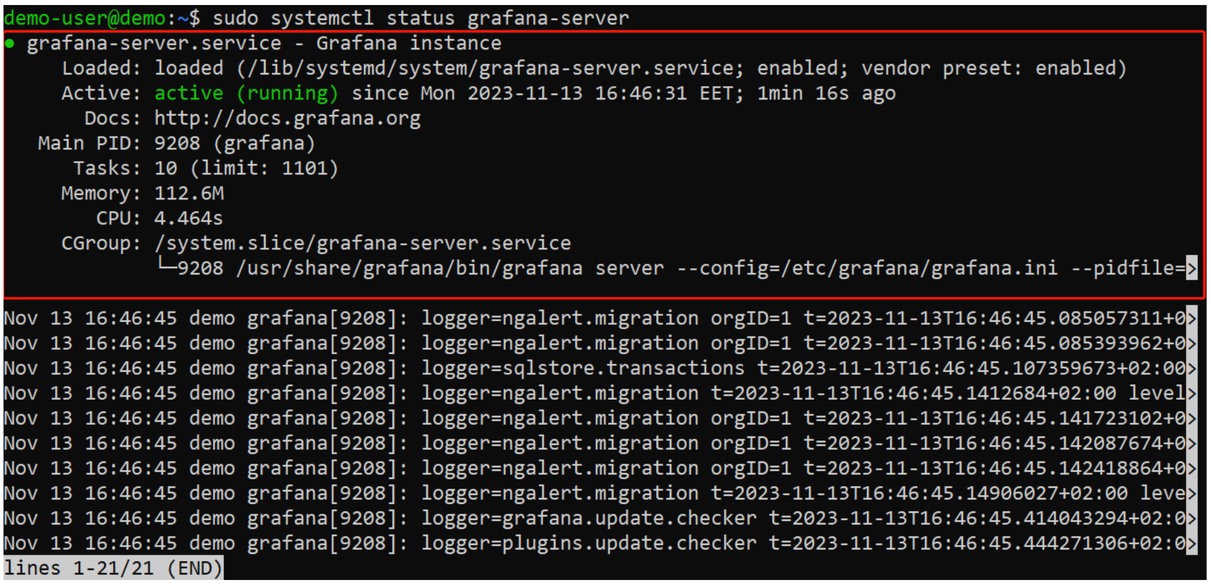


```
demo-user@demo:~$ sudo grafana-server -v
Version 10.2.0 (commit: 895fbafb7a, branch: HEAD)
demo-user@demo:~$ sudo systemctl enable grafana-server
Synchronizing state of grafana-server.service with SysV service script with /lib/systemd/systemd-sysv-i
nstall.
Executing: /lib/systemd/systemd-sysv-install enable grafana-server
Created symlink /etc/systemd/system/multi-user.target.wants/grafana-server.service → /lib/systemd/syste
m/grafana-server.service.
demo-user@demo:~$ sudo systemctl start grafana-server
```

#Step 7 - Verify that the Grafana service is running

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

If the Grafana service was started successfully, you should see a sign that it is active and running.

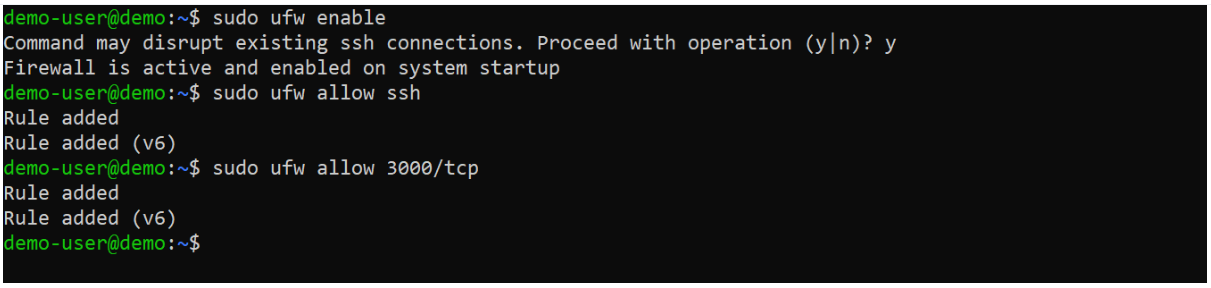


```
demo-user@demo:~$ sudo systemctl status grafana-server
● grafana-server.service - Grafana instance
     Loaded: loaded (/lib/systemd/system/grafana-server.service; enabled; vendor preset: enabled)
     Active: active (running) since Mon 2023-11-13 16:46:31 EET; 1min 16s ago
       Docs: http://docs.grafana.org
   Main PID: 9208 (grafana)
      Tasks: 10 (limit: 1101)
     Memory: 112.6M
        CPU: 4.464s
     CGroup: /system.slice/grafana-server.service
             └─9208 /usr/share/grafana/bin/grafana server --config=/etc/grafana/grafana.ini --pidfile=>

Nov 13 16:46:45 demo grafana[9208]: logger=ngalert.migration orgID=1 t=2023-11-13T16:46:45.085057311+0>
Nov 13 16:46:45 demo grafana[9208]: logger=ngalert.migration orgID=1 t=2023-11-13T16:46:45.085393962+0>
Nov 13 16:46:45 demo grafana[9208]: logger=sqlstore.transactions t=2023-11-13T16:46:45.107359673+02:00>
Nov 13 16:46:45 demo grafana[9208]: logger=ngalert.migration t=2023-11-13T16:46:45.1412684+02:00 level>
Nov 13 16:46:45 demo grafana[9208]: logger=ngalert.migration orgID=1 t=2023-11-13T16:46:45.141723102+0>
Nov 13 16:46:45 demo grafana[9208]: logger=ngalert.migration orgID=1 t=2023-11-13T16:46:45.142087674+0>
Nov 13 16:46:45 demo grafana[9208]: logger=ngalert.migration orgID=1 t=2023-11-13T16:46:45.142418864+0>
Nov 13 16:46:45 demo grafana[9208]: logger=ngalert.migration t=2023-11-13T16:46:45.14906027+02:00 leve>
Nov 13 16:46:45 demo grafana[9208]: logger=grafana.update.checker t=2023-11-13T16:46:45.414043294+02:0>
Nov 13 16:46:45 demo grafana[9208]: logger=plugins.update.checker t=2023-11-13T16:46:45.444271306+02:0>
lines 1-21/21 (END)
```

#Step 8 - Open the port in the firewall

```
sudo ufw enable
sudo ufw allow ssh
sudo ufw allow 3000/tcp
```

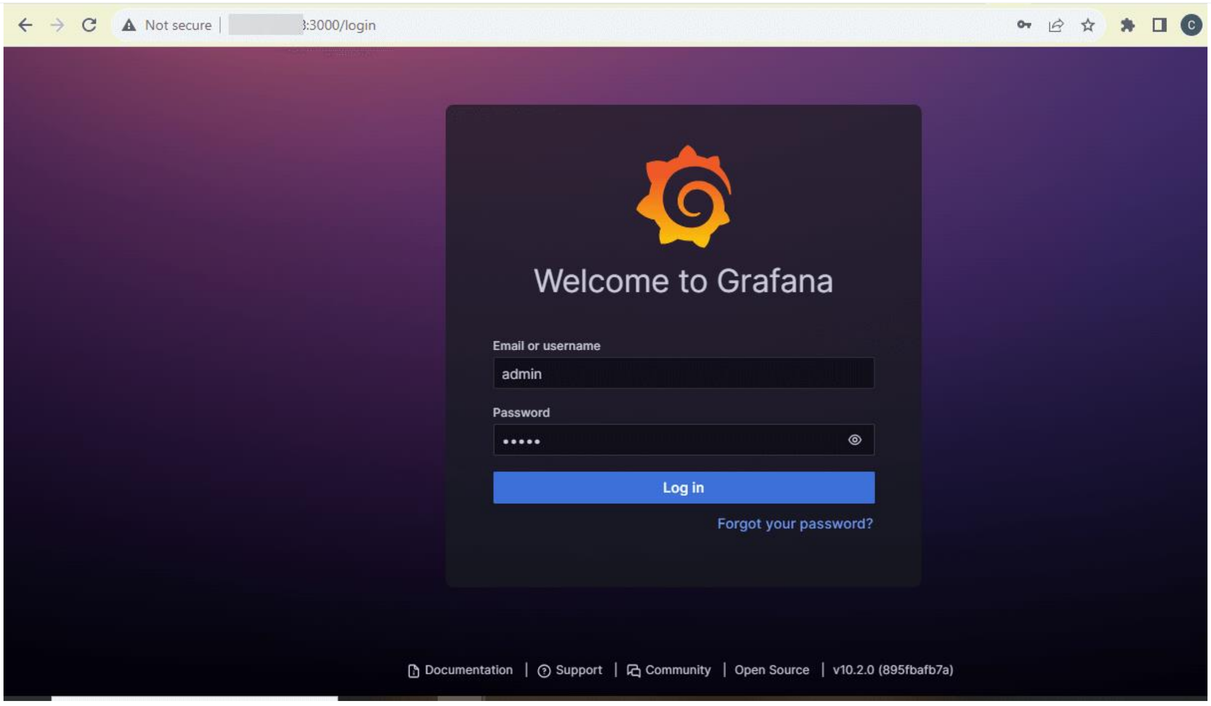


```
demo-user@demo:~$ sudo ufw enable
Command may disrupt existing ssh connections. Proceed with operation (y|n)? y
Firewall is active and enabled on system startup
demo-user@demo:~$ sudo ufw allow ssh
Rule added
Rule added (v6)
demo-user@demo:~$ sudo ufw allow 3000/tcp
Rule added
Rule added (v6)
demo-user@demo:~$
```

#Step 9 - Access the Grafana web interface

To access the Grafana web interface, open a web browser and enter the IP address of your server (or hostname if applicable), followed by port 3000. The URL format should be `http://your_server_IP:3000`. Once loaded, you should see the Grafana login page. The default credentials are:

- **Username: admin**
- **Password: admin**



Not secure | [redacted]:3000/login

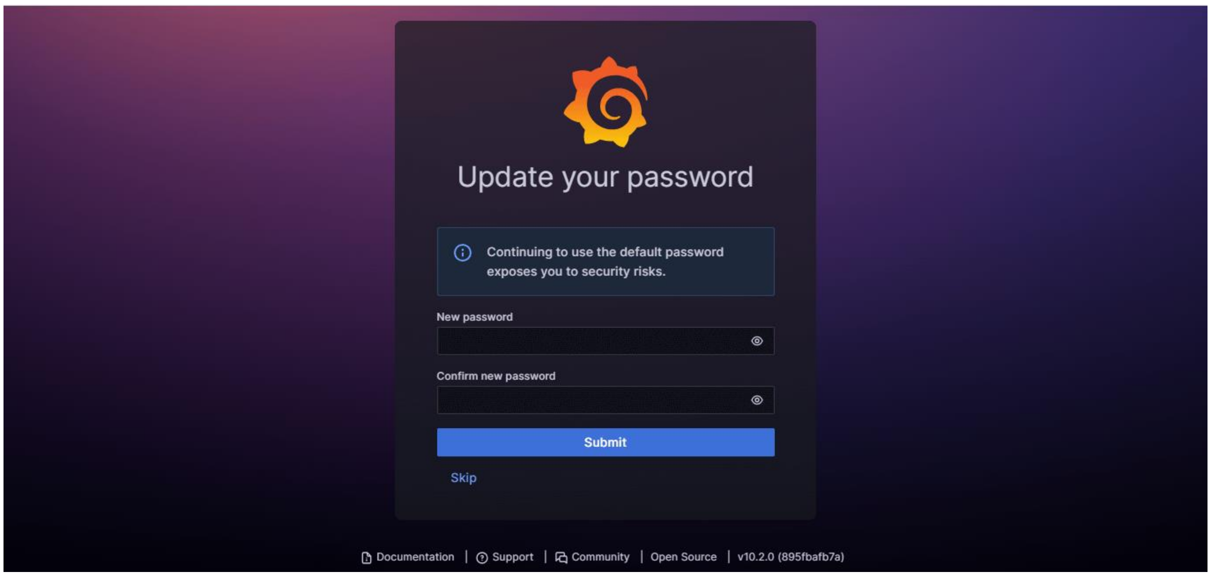
Welcome to Grafana

Email or username: admin

Password: •••••

Log in
Forgot your password?
Documentation | Support | Community | Open Source | v10.2.0 (895fbafb7a)

You'll be prompted to create a new password. Input a secure password, confirm it, and click the "Submit" button.



Update your password

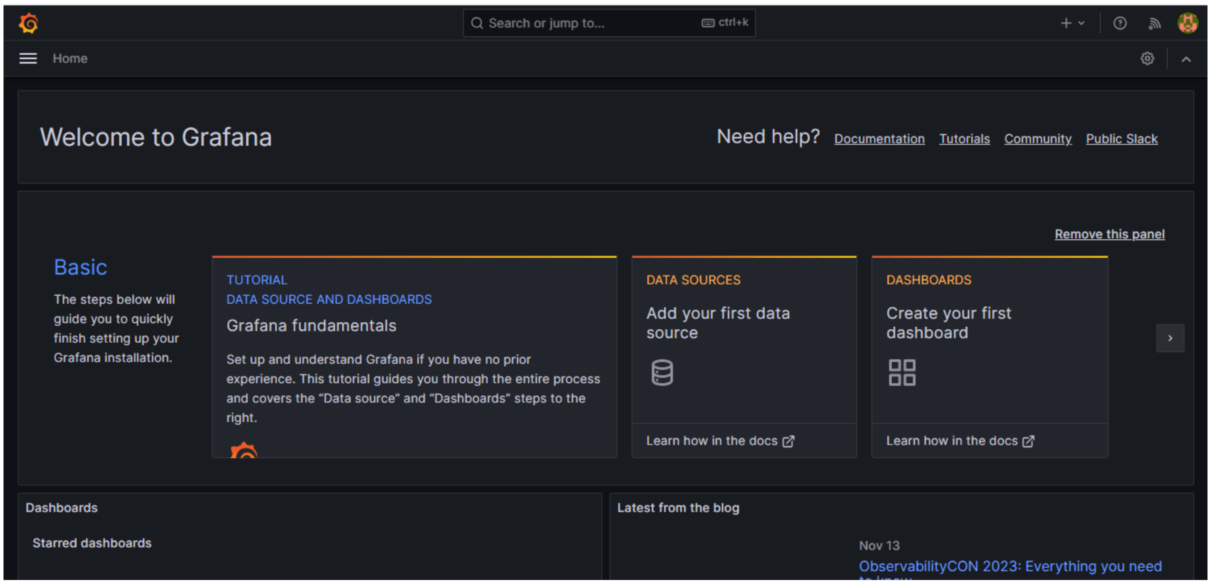
Continuing to use the default password exposes you to security risks.

New password

Confirm new password

Submit
Skip
Documentation | Support | Community | Open Source | v10.2.0 (895fbafb7a)

Once done, you'll have access to Grafana's dashboard.



Welcome to Grafana

Need help? Documentation Tutorials Community Public Slack

Basic

The steps below will guide you to quickly finish setting up your Grafana installation.

TUTORIAL
DATA SOURCE AND DASHBOARDS
Grafana fundamentals

Set up and understand Grafana if you have no prior experience. This tutorial guides you through the entire process and covers the "Data source" and "Dashboards" steps to the right.

DATA SOURCES
Add your first data source

Learn how in the docs

DASHBOARDS
Create your first dashboard

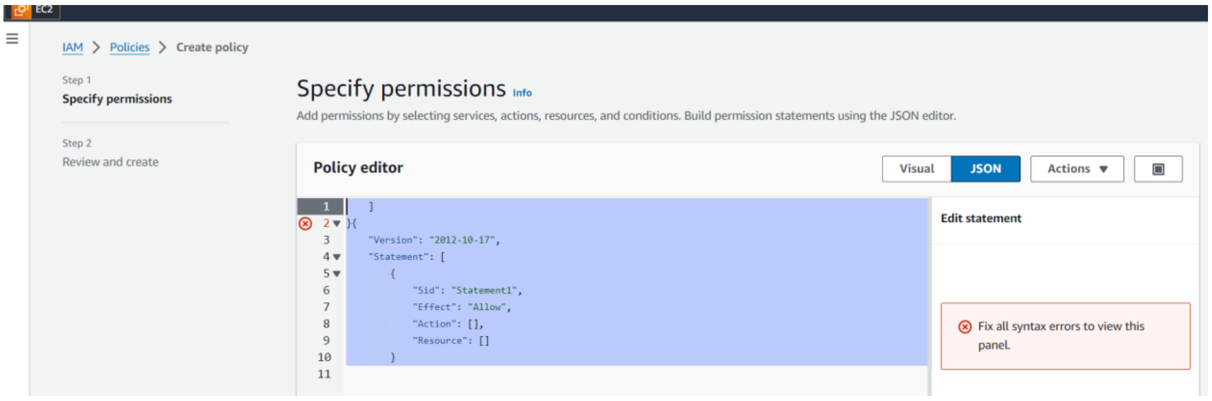
Learn how in the docs

Latest from the blog

Nov 13
ObservabilityCON 2023: Everything you need

/

Now open the AWS console and select the option IAM. first, Click on the **JSON** tab, Remove the existing code, and copy-paste the below policy



Specify permissions Info

Add permissions by selecting services, actions, resources, and conditions. Build permission statements using the JSON editor.

Policy editor

```
]
}{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Allow",
            "Action": [],
            "Resource": []
        }
```

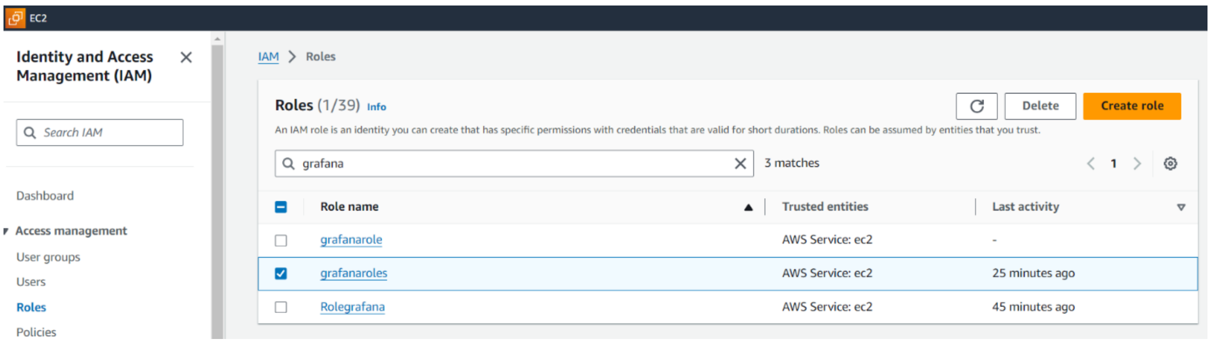
Fix all syntax errors to view this panel.

statement into the editor:

Policy:

```
{
        "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":"*"
        }
        ]
}
```

Policy is created. Now click on the role tab. select the create role. Choose **Service or use case: EC2** click on the next. Select the policy and create a role. Your role is created-



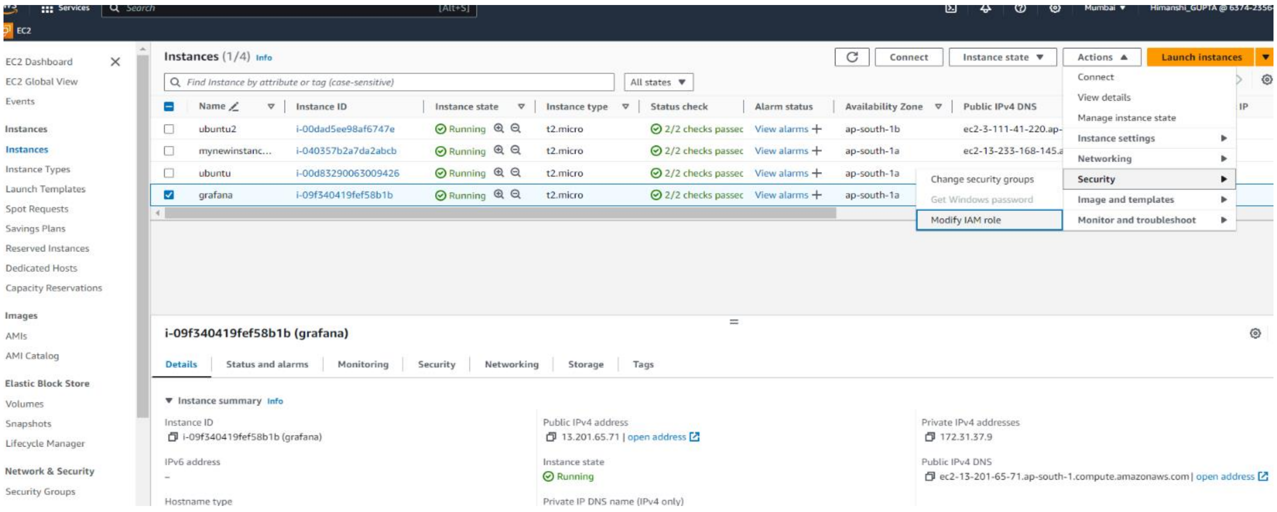
Roles (1/39) Info

An IAM role is an identity you can create that has specific permissions with credentials that are valid for short durations. Roles can be assumed by entities that you trust.

grafana — 3 matches

	Role name	Trusted entities	Last activity
☐	grafanarole	AWS Service: ec2	-
☑	grafanaroles	AWS Service: ec2	25 minutes ago
☐	Rolegrafana	AWS Service: ec2	45 minutes ago

Now go back to the Instances. Select your instance, click on the action tab, select the security, and then click on the Modify IAM role.



Instances (1/4)

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IPv4 DNS
☐	ubuntu2	i-00dad5ee98af6747e	Running	t2.micro	2/2 checks passec	View alarms	ap-south-1b	ec2-3-111-41-220.ap-
☐	mynewinstanc...	i-040357b2a7da2abcb	Running	t2.micro	2/2 checks passec	View alarms	ap-south-1a	ec2-13-233-168-145.a
☐	ubuntu	i-00d83290063009426	Running	t2.micro	2/2 checks passec	View alarms	ap-south-1a	
☑	grafana	i-09f340419fef58b1b	Running	t2.micro	2/2 checks passec	View alarms	ap-south-1a	

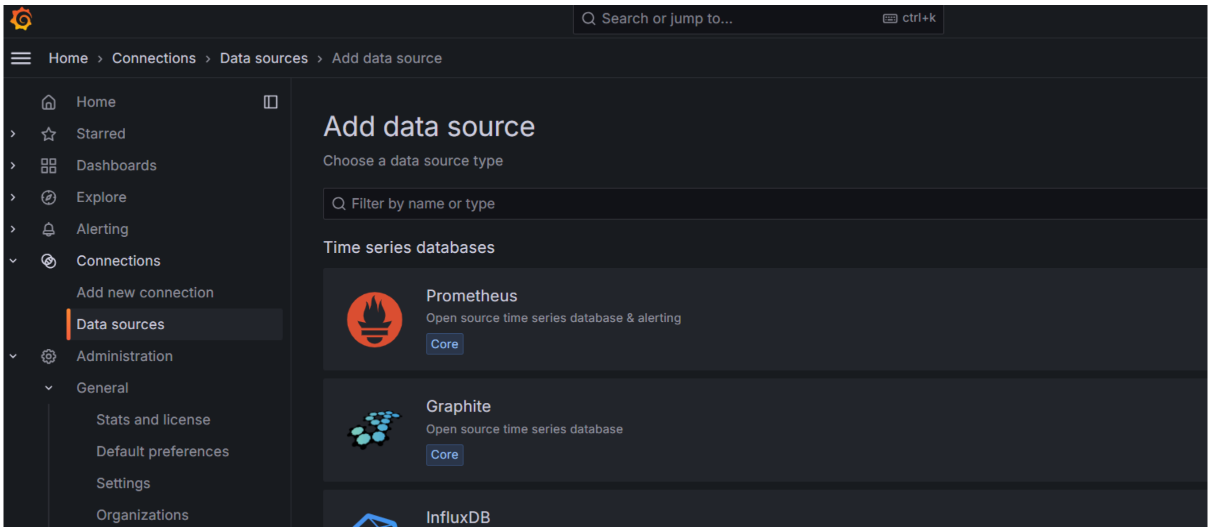
i-09f340419fef58b1b (grafana)

Instance summary

Instance ID: i-09f340419fef58b1b (grafana)
Public IPv4 address: 13.201.65.71 | open address
Private IPv4 addresses: 172.31.37.9
IPv6 address: –
Instance state: Running
Public IPv4 DNS: ec2-13-201-65-71.ap-south-1.compute.amazonaws.com | open address
Hostname type
Private IP DNS name (IPv4 only)

Select your role then click on update IAM role. now it’s done to see your graph.

Go back to the Grafana dashboard. click on the connection tab and select the data source option.



Add data source

Choose a data source type

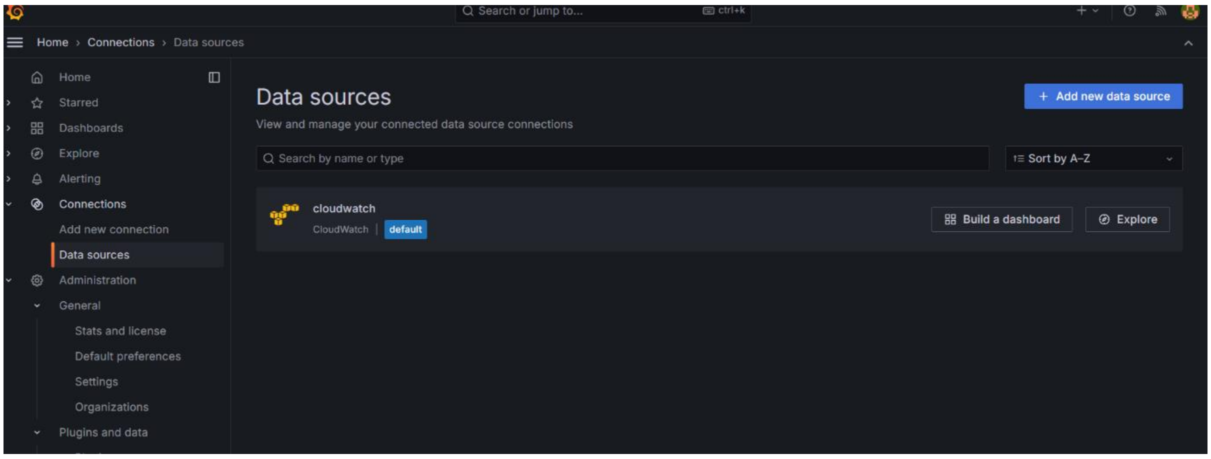
Time series databases

Prometheus
Open source time series database & alerting
Core

Graphite
Open source time series database
Core

InfluxDB

Select the cloud watch.

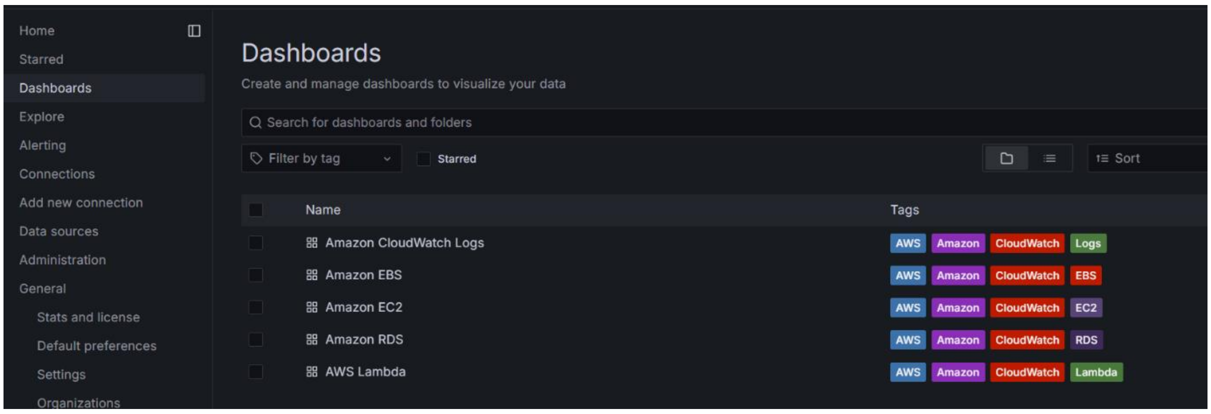


Data sources

View and manage your connected data source connections

Name	Type	
cloudwatch	CloudWatch	default

Check it in the Dashboard. You see your screen like that-

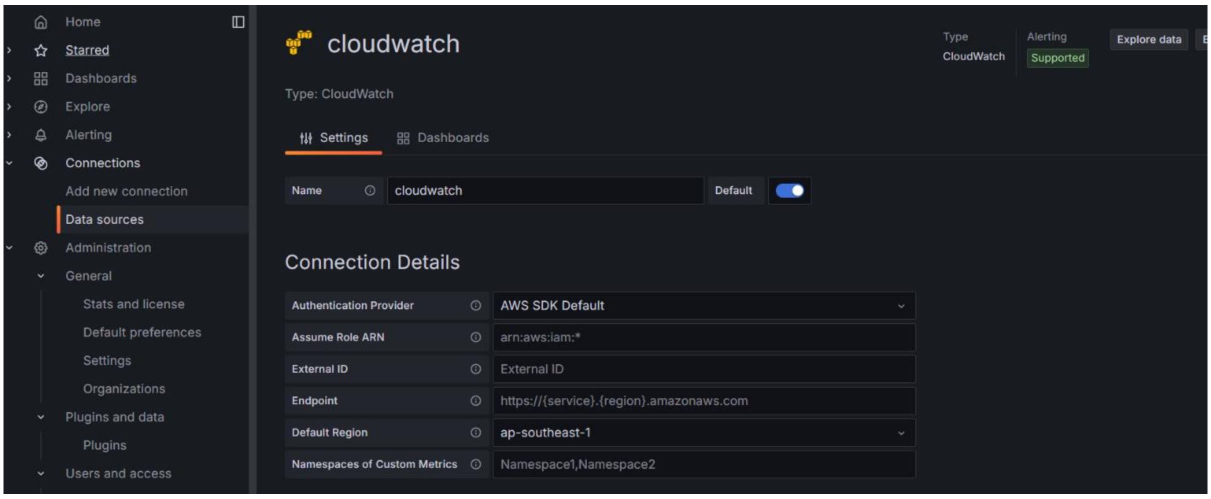


Dashboards

Create and manage dashboards to visualize your data

Name	Tags
Amazon CloudWatch Logs	AWS Amazon CloudWatch Logs
Amazon EBS	AWS Amazon CloudWatch EBS
Amazon EC2	AWS Amazon CloudWatch EC2
Amazon RDS	AWS Amazon CloudWatch RDS
AWS Lambda	AWS Amazon CloudWatch Lambda

Click on the Amazon ec2 you see some errors.so click on the data source and change the region **ap-southeast-1** like that -



cloudwatch

Type: CloudWatch

Type: CloudWatch | Alerting: Supported

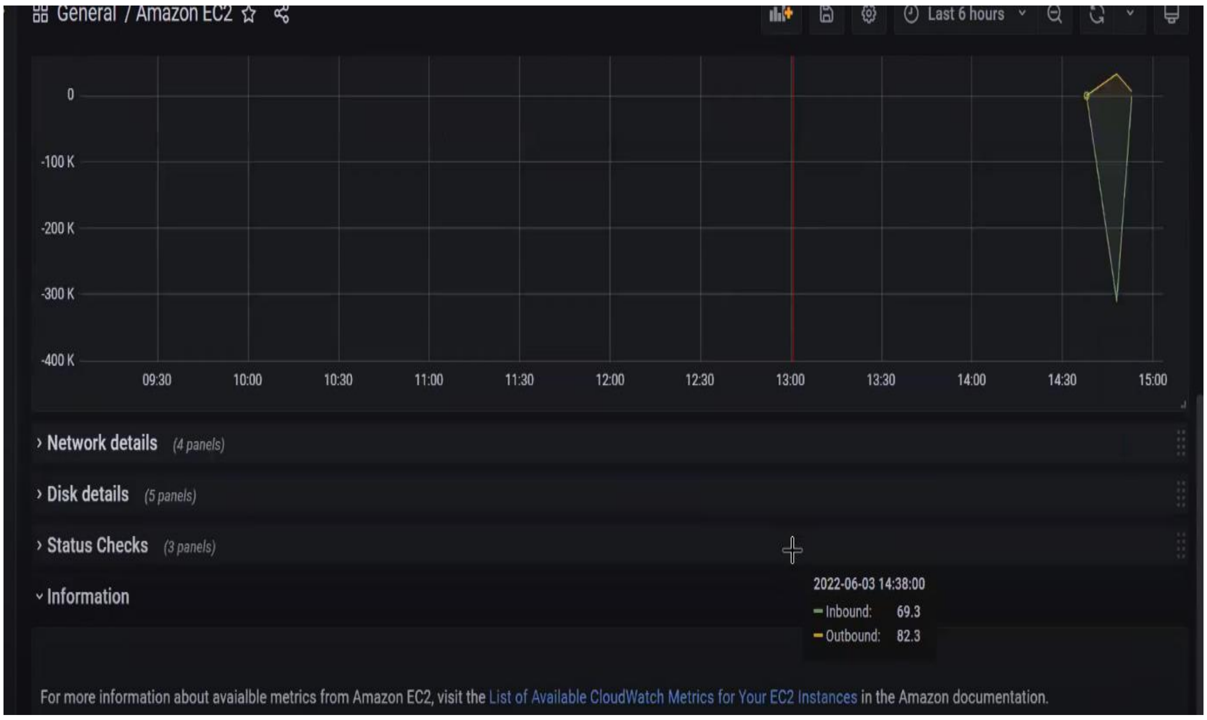
Name: cloudwatch | Default: on

Connection Details

Authentication Provider	AWS SDK Default
Assume Role ARN	arn:aws:iam:*
External ID	External ID
Endpoint	https://{service}.{region}.amazonaws.com
Default Region	ap-southeast-1
Namespaces of Custom Metrics	Namespace1,Namespace2

Save those changes.

Go back to the Dashboard now you see your Grafana graph.



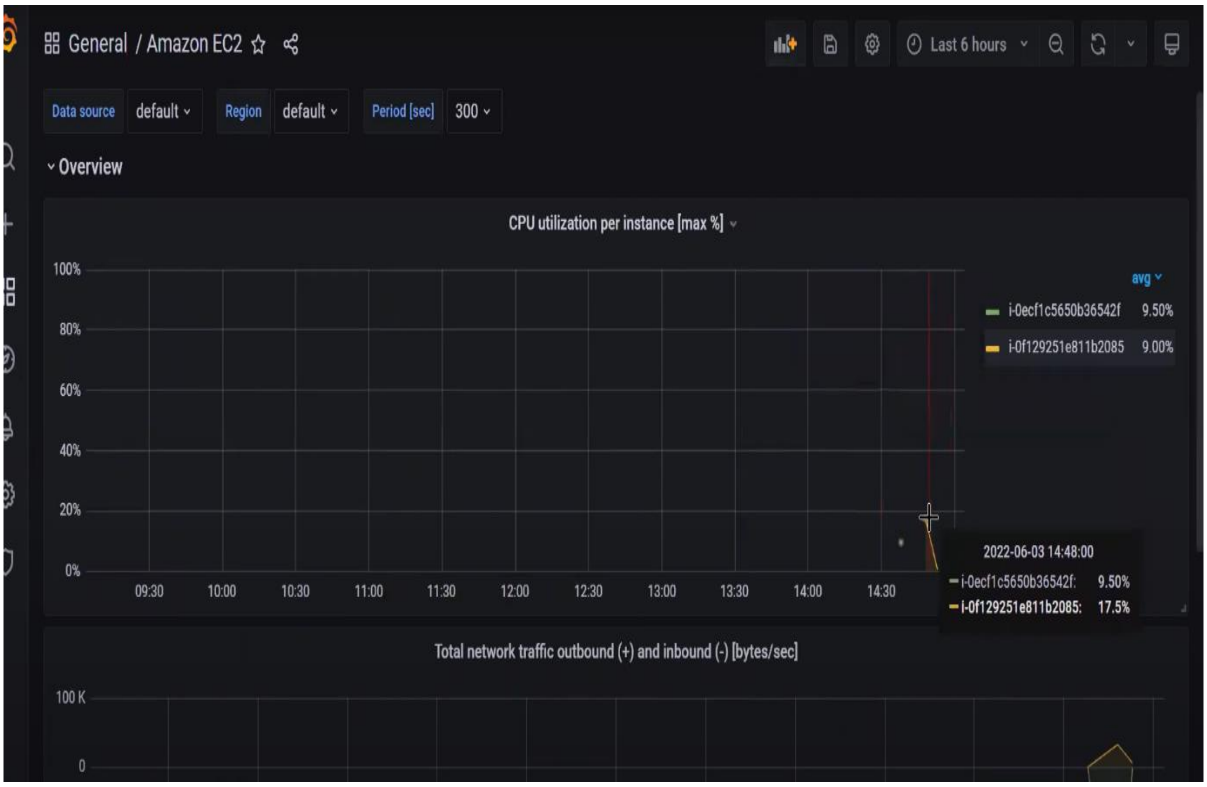
› Network details (4 panels)

› Disk details (5 panels)

› Status Checks (3 panels)

⌄ Information

For more information about avaialble metrics from Amazon EC2, visit the List of Available CloudWatch Metrics for Your EC2 Instances in the Amazon documentation.



Data source default · Region default · Period [sec] 300

⌄ Overview

CPU utilization per instance [max %]

i-0ecf1c5650b36542f	9.50%
i-0f129251e811b2085	9.00%

i-0ecf1c5650b36542f: 9.50%
i-0f129251e811b2085: 17.5%

Total network traffic outbound (+) and inbound (-) [bytes/sec]