

IoT sensor dashboard

Deployment Guide

[Author name]
7-25-2023

1. Introduction

This is an accelerator for creating an IoT solution to view data from sensors, predominantly LoRaWAN sensors, using AWS IoT services. It should be noted that since this is an accelerator, it is designed to be a starting point of your solution build journey rather than a complete solution. However, this accelerator does create all the AWS resources required to interface with LoRaWAN gateways, ingest, decode & store LoRaWAN messages, along with template dashboards for the supported types of sensors. It also supports ingesting data from any MQTT sensor which can connect and publish to IoT Core directly. Therefore, it can be used to cater to certain basic use cases with minimal customization.

This accelerator utilizes the following key AWS services:

- AWS IoT Core – IoT Core acts as the connection point for your MQTT based sensors to the AWS cloud. IoT Core for LoRaWAN provides a fully managed LoRaWAN Network Server (LNS) that enables customers to connect wireless devices that use the LoRaWAN protocol
- AWS Lambda – There are 2 Lambda functions in this accelerator
 - Decoder function – this function decodes the uplink data from the sensors based on the sensor's vendor and the model. This is to decode uplink messages from LoRaWAN sensors.
 - Sensor list update function – this function updates the database table with IoT thing create and delete events to maintain a table of provisioned sensors
- Amazon Timestream – this is where the timeseries data transmitted from the sensors are stored. It also contains a separate table to maintain the list of provisioned sensors along with vendor and model information for dashboarding purposes.
- Amazon Managed Grafana – this is used as the dashboarding layer. Users can consume the prebuilt dashboards for each model of sensor or build their own dashboards while using the given dashboards as a reference.

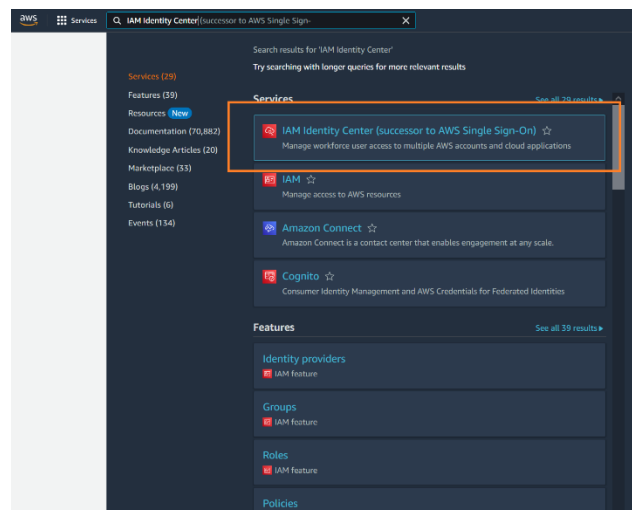
In the background, AWS IAM Identity Center provides the single sign on for the Amazon Managed Grafana, thus allowing the administrator the flexibility to segregate the dashboard logins from AWS account logins. It also allows the end user to utilize the same single sign on that the user might be already using on other AWS services.

2. Deployment guide

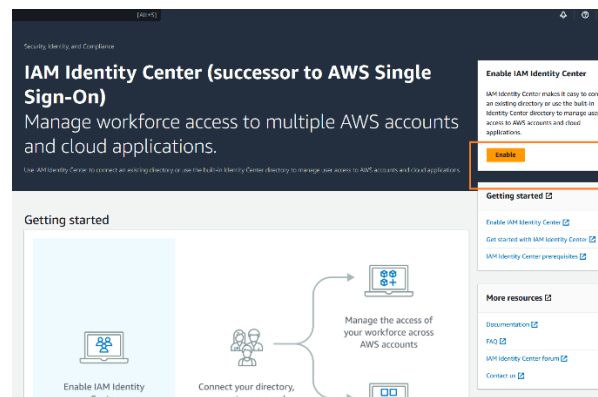
Step 1 – Enable IAM Identity Center and create a user

Important: This step can be skipped if IAM Identity Center is enabled and users already exist. If this is a fresh AWS account, follow the instructions below.

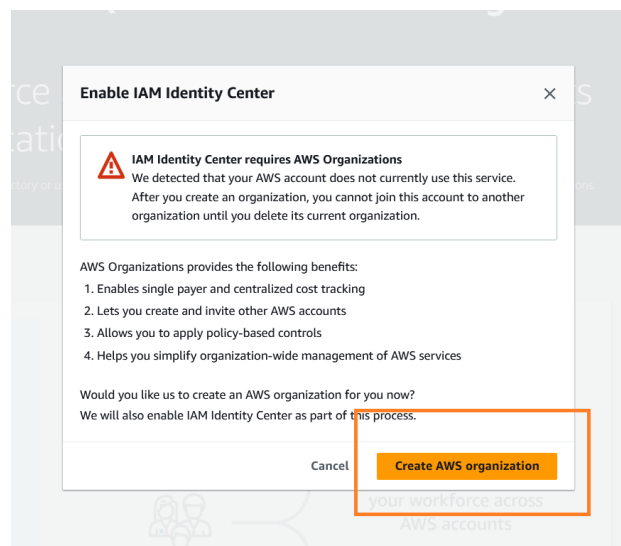
1.1 Open IAM Identity Center



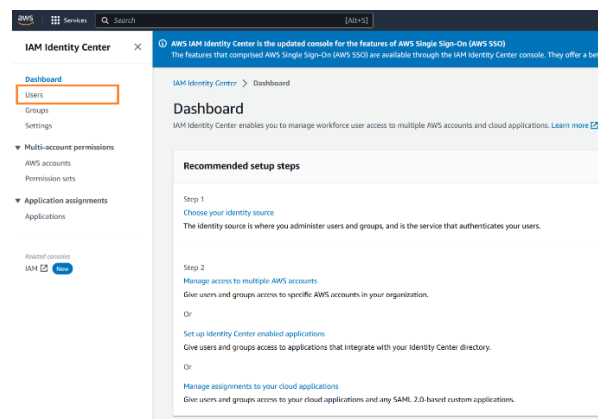
1.2 Enable IAM Identity Center



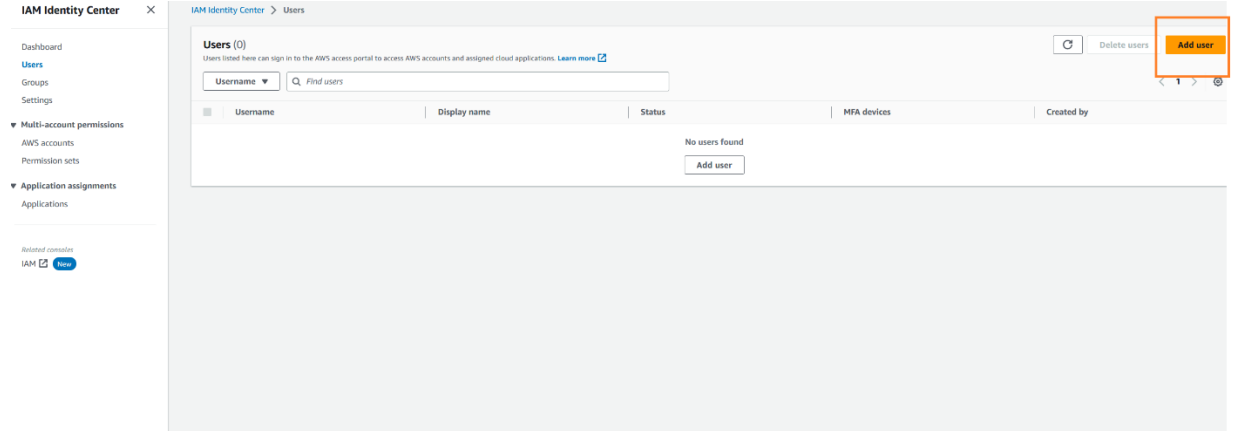
1.3 Confirm that you want to create an AWS Org.



1.4 Navigate to Users



1.5 Add a User



IAM Identity Center X

IAM Identity Center > Users

Users (0) 🔄 Delete users **Add user**

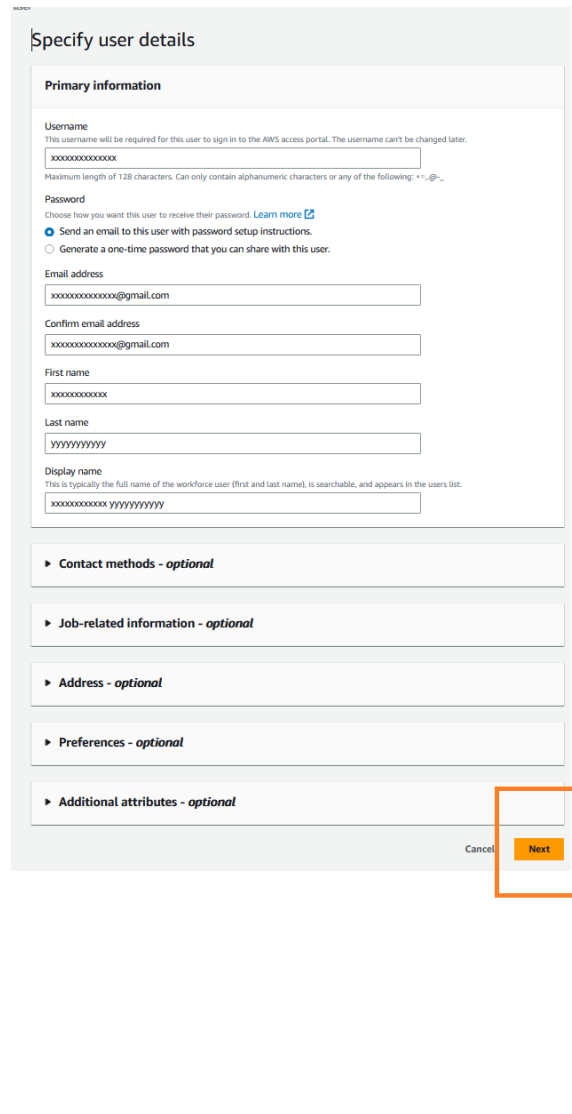
Users listed here can sign in to the AWS access portal to access AWS accounts and assigned cloud applications. [Learn more](#)

Username Find users

Username	Display name	Status	MFA devices	Created by
No users found				

Add user

1.5.1 Add a User (contd.)



Specify user details

Primary information

Username
This username will be required for this user to sign in to the AWS access portal. The username can't be changed later.
 xxxxxxxxxxxxxxxx
Maximum length of 128 characters. Can only contain alphanumeric characters or any of the following: +, -, @, _.

Password
Choose how you want this user to receive their password. [Learn more](#)
☒ Send an email to this user with password setup instructions.
☐ Generate a one-time password that you can share with this user.

Email address
 xxxxxxxxxxxxxxxx@gmail.com

Confirm email address
 xxxxxxxxxxxxxxxx@gmail.com

First name
 xxxxxxxxxxxxxxxx

Last name
 yyyyyyyyyyy

Display name
This is typically the full name of the workforce user (first and last name), is searchable, and appears in the users list.
 xxxxxxxxxxxx yyyyyyyyyyy

Contact methods - optional

Job-related information - optional

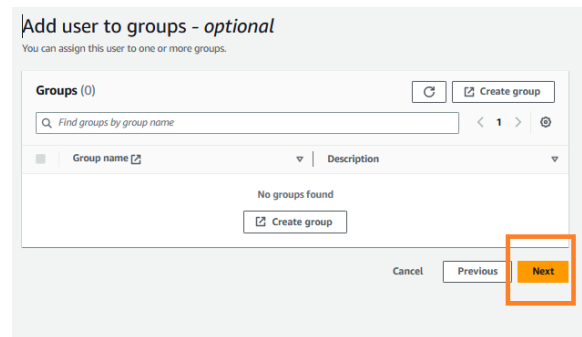
Address - optional

Preferences - optional

Additional attributes - optional

Cancel **Next**

1.5.3 Add a User (contd.)



Add user to groups - optional
You can assign this user to one or more groups.

Groups (0) 🔄 Create group

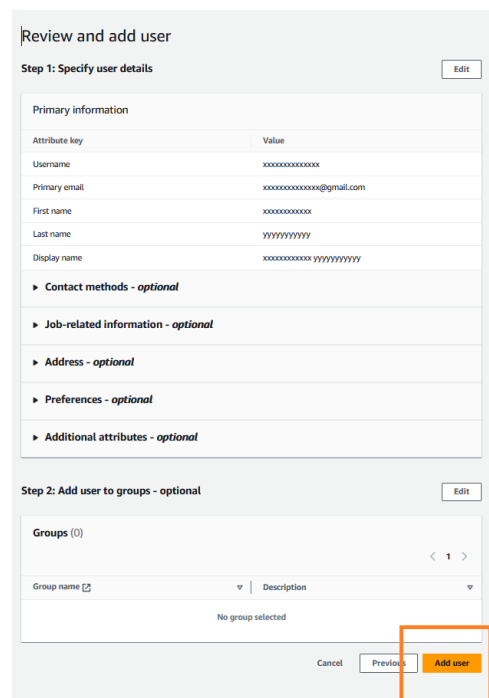
Find groups by group name

Group name	Description
No groups found	

Create group

Cancel Previous **Next**

1.5.3 Add a User (contd.)



Review and add user Edit

Step 1: Specify user details

Primary information

Attribute key	Value
Username	xxxxxxxxxxxxx
Primary email	xxxxxxxxxxxxx@gmail.com
First name	xxxxxxxxxxxxx
Last name	yyyyyyyyyy
Display name	xxxxxxxxxxxxx yyyyyyyyyyy

Contact methods - optional

Job-related information - optional

Address - optional

Preferences - optional

Additional attributes - optional

Step 2: Add user to groups - optional Edit

Groups (0) < 1 >

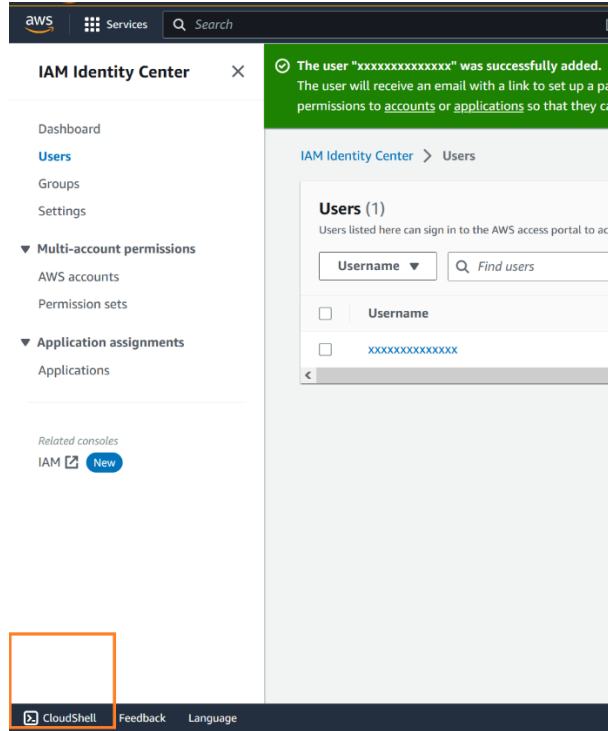
Group name	Description
No group selected	

Cancel Previous **Add user**

Step 2 – Create stack

This step will create the entire stack in your AWS account with a single command in the AWS cloud shell

2.1 Open AWS Cloud Shell



2.2 Enter the following command:

```
git clone git@github.com:aws-samples/lorawandash.git && cd lorawandash && python3 setup_stack.py
```



2.3 Stack creation in progress

```
AWS CloudShell

us-east-1

[cloudshell-user@ip-10-6-14-170 ~]$ git clone git@ssh.gitlab.aws.dev:akalanka/lorawandash.git && cd lorawandash && python3 setup_stack.py
Cloning into 'lorawandash'...
The authenticity of host 'ssh.gitlab.aws.dev (52.54.97.5)' can't be established.
ECDSA key fingerprint is SHA256:pJONIKSKW4dsBr8gbZIsPctNNHyMjfdPefNrbFkGh40.
ECDSA key fingerprint is MD5:e9:d8:95:08:fe:65:61:bb:ab:6b:ba:c7:3c:8c:2f:b0.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'ssh.gitlab.aws.dev,52.54.97.5' (ECDSA) to the list of known hosts.
remote: Enumerating objects: 46, done.
remote: Counting objects: 100% (46/46), done.
remote: Compressing objects: 100% (36/36), done.
remote: Total 46 (delta 14), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (46/46), 20.33 KiB | 991.00 KiB/s, done.
Resolving deltas: 100% (14/14), done.
1. Creating stack : lorawandash
{'StackId': 'arn:aws:cloudformation:us-east-1:461753770371:stack/lorawandash/b8c764c0-2a8b-11ee-b3a7-124b3c2fe307', 'ResponseMetadata': {'RequestId': 'a53f4729-b2e5-45a3-b89e-c13529ca915f', 'date': 'Tue, 25 Jul 2023 01:36:48 GMT', 'content-type': 'text/xml', 'content-length': '381', 'connection': 'keep-alive'}}
CREATE_IN_PROGRESS
```

2.4 Stack Creation complete

```
AWS CloudShell

us-east-1

.
.
CREATE_COMPLETE
2. Stack created
3.1 Updating decoder lambda function: lorawandash-DefaultDecoderLambda-j6mwDg15bCCF
Zipping source files
  adding: index.js (deflated 62%)
  adding: vendors/ (stored 0%)
  adding: vendors/Netvox/ (stored 0%)
  adding: vendors/Netvox/RB11E.js (deflated 46%)
  adding: vendors/Netvox/R718A.js (deflated 45%)
0
Decoder lambda function updated successfully
3.2 Updating list lambda function: lorawandash-ListLambda-rgiS40aM2Kyc
Zipping source files
  adding: index.js (deflated 65%)
0
List lambda function updated successfully
4. Enabling thing event notifications in IoT Core
5. Creating template dashboards
Data source created
Created folder Samples-Sensor
Creating dashboards in folder Samples-Sensor
Created dashboard: RB11E.json
Created dashboard: R718A.json

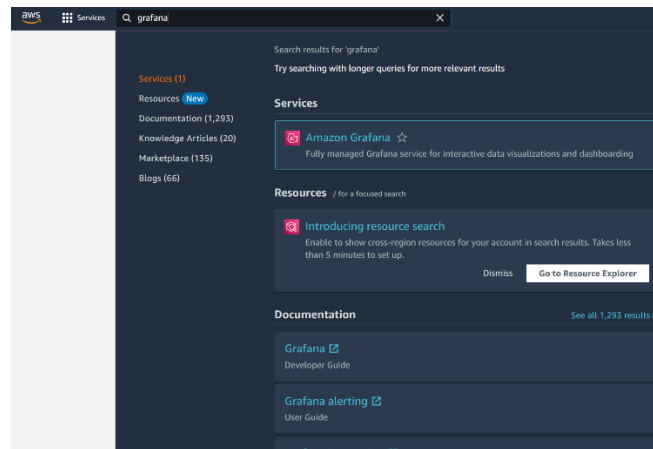
Next steps: >>>>
1. Navigate to the Amazon Managed Grafana page in the AWS console and click on the workspace name
2. Assign a user under Authentication > AWS IAM Identity Center
3. Make the user an Admin
4. Login to your Grafana URL: https://g-5bf30ad076.grafana-workspace.us-east-1.amazonaws.com with the credentials for the assigned user and start exploring!
[cloudshell-user@ip-10-6-14-170 lorawandash]$
```

At the end of the stack creation it will provide the next steps to assign a user to the Grafana instance which is described in step 3.

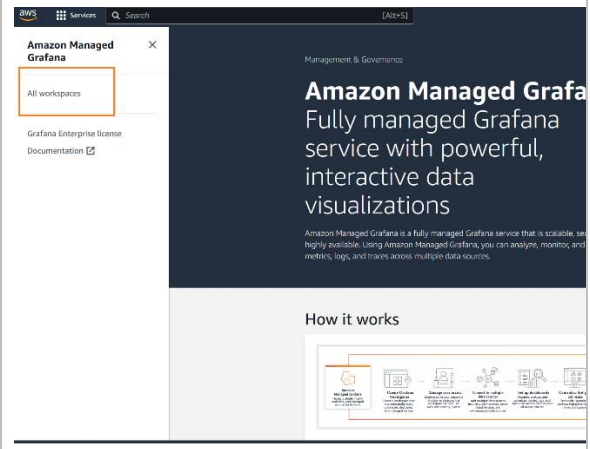
Step 3 – Assign Users to Grafana

This step will assign users to the Grafana workspace. Ensure that the users are created in IAM Identity Center by following Step 1. Furthermore ensure that the email for newly created users are verified and the password is set.

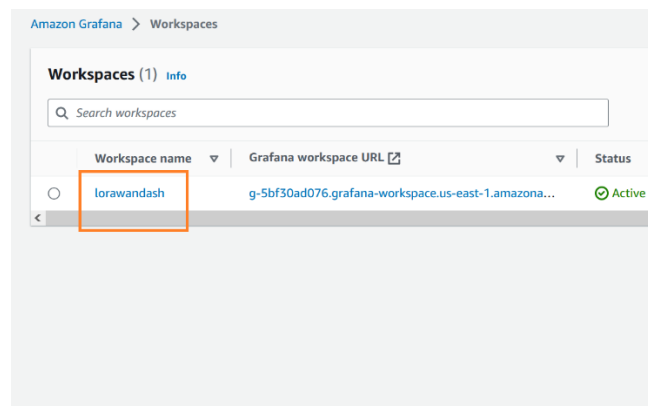
3.1 Navigate to Amazon Managed Grafana



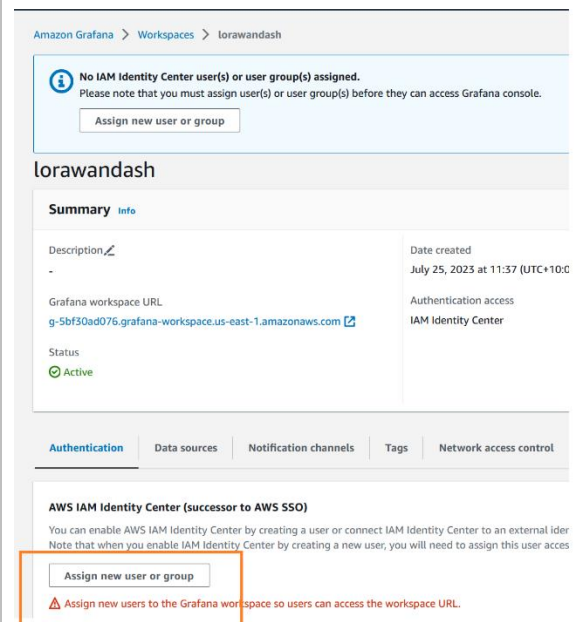
3.2 Open All Workspaces



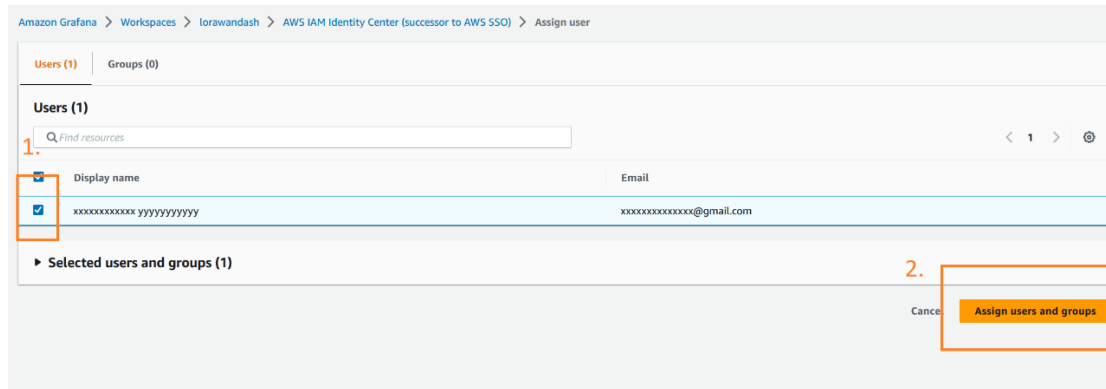
3.3 Open the workspace created in step 2.



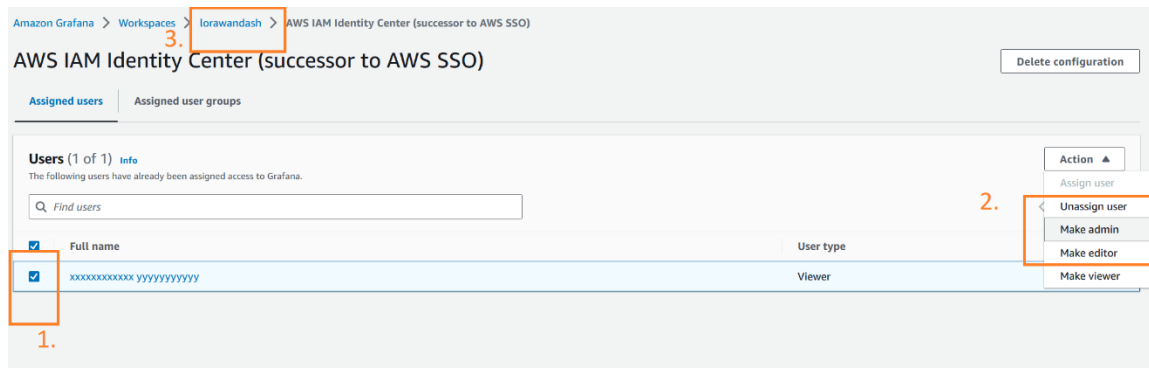
3.4 Assign a user



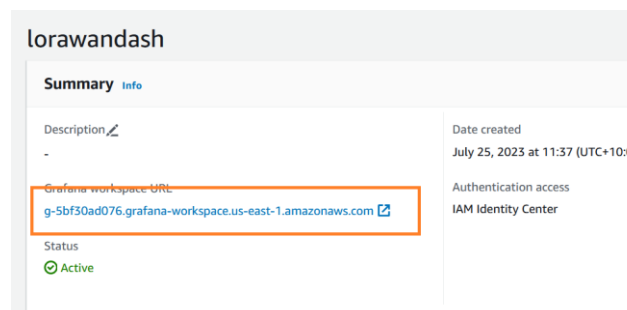
3.4 Assign a user (contd.)



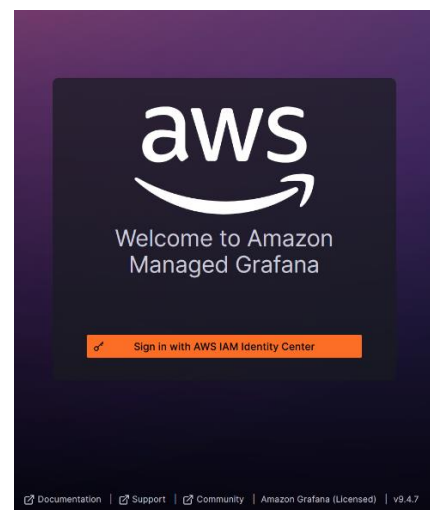
3.5 Make the user and admin/viewer/editor as desired and go back to the workspace



3.6 Click to open the Grafana workspace



3.7 Login with the created user credentials



This concludes the stack creation process. For further details on the operational aspects, please refer to the user guide.