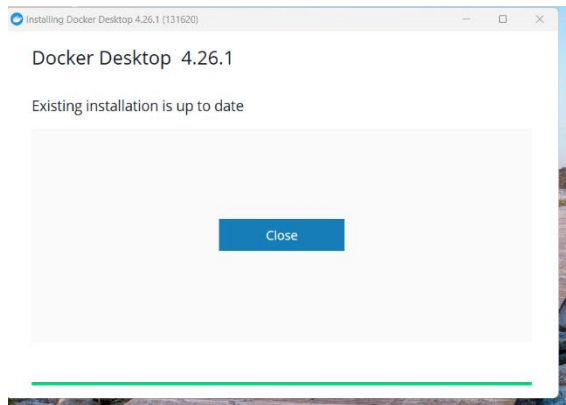
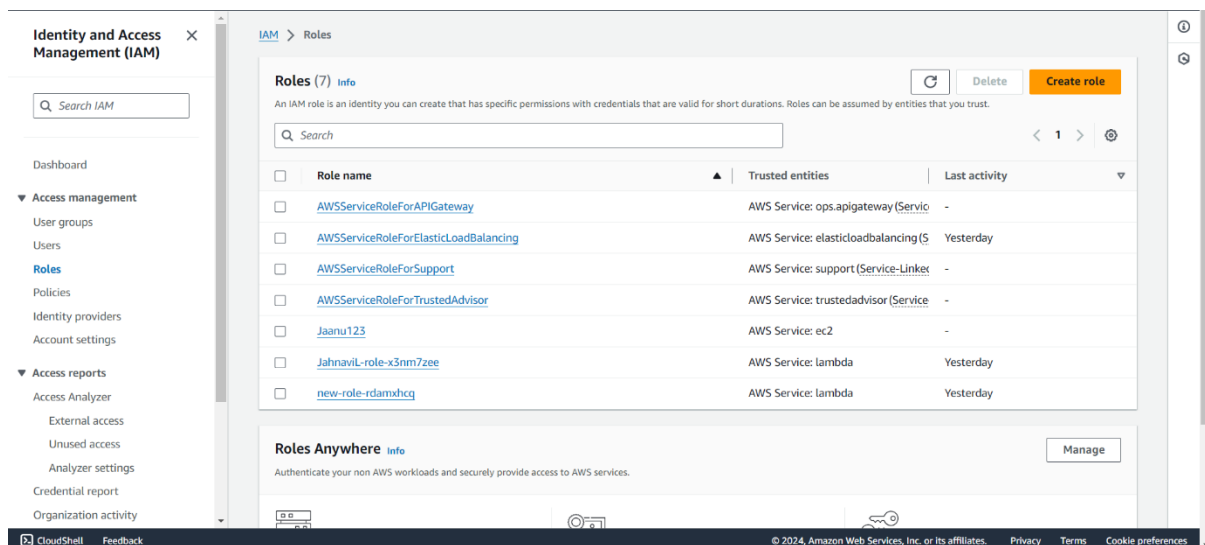


Day 19

1) Setup Docker



2) Setup IAM Role to copy folder using crontab from ec2 to s3



aws

Services

Search

[Alt+S]

Global

Jahnavi Yarlagadda

Identity and Access Management (IAM)

Search IAM

Dashboard

Access management

Users

Roles

Policies

Identity providers

Account settings

Access reports

Access Analyzer

External access

Unused access

Analyzer settings

Credential report

Organization activity

IAM > Roles > Jaanu123

Jaanu123

Info

Delete

Allows EC2 instances to call AWS services on your behalf.

Summary

Edit

Creation date	ARN	Instance profile ARN
February 01, 2024, 09:17 (UTC-08:00)	arn:aws:iam::891377370771:role/Jaanu123	arn:aws:iam::891377370771:instance-profile/Jaanu123
Last activity	Maximum session duration	
-	1 hour	

Permissions

Trust relationships

Tags

Access Advisor

Revoke sessions

Permissions policies (1)

Info

Simulate

Remove

Add permissions

You can attach up to 10 managed policies.

Search

Filter by Type

All types

Policy name

Type

Attached entities

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Services

Search

[Alt+S]

Ohio

Jahnavi Yarlagadda

EC2 > Instances > i-0b703d3944467e26b > Modify IAM role

Modify IAM role

Info

Attach an IAM role to your instance.

Instance ID

i-0b703d3944467e26b (Test2)

IAM role

Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.

Jaanu123

Create new IAM role

Cancel

Update IAM role

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates. Privacy Terms Cookie preferences

aws

Services

Search

[Alt+S]

Ohio

Jahnavi Varlagadda

EC2 Dashboard

EC2 Global View

Events

Instances

Instances

Instance Types

Launch Templates

Spot Requests

Savings Plans

Reserved Instances

Dedicated Hosts

Capacity Reservations

New

Images

AMIs

AMI Catalog

Elastic Block Store

Volumes

Snapshots

Lifecycle Manager

Successfully attached Jaanu123 to instance i-0b703d3944467e26b

Instances (1) Info

Find Instance by attribute or tag (case-sensitive)

Any state

Connect

Instance state

Actions

Launch instances

	Name	Instance ID	Instance state	Instance type	Status check	Alarm status	Availability Zone	Public IP
<input type="checkbox"/>	Test2	i-0b703d3944467e26b	Running	t2.micro	2/2 checks passed	View alarms	us-east-2b	ec2-18-111

Select an instance

CloudShell

Feedback

© 2024, Amazon Web Services, Inc. or its affiliates.

Privacy

Terms

Cookie preferences

aws

Services

Search

[Alt+S]

Global

Jahnavi Varlagadda

Successfully created bucket "jaanu456"

To upload files and folders, or to configure additional bucket settings, choose View details.

View details

Amazon S3

Buckets

Account snapshot

Storage lens provides visibility into storage usage and activity trends. Learn more

View Storage Lens dashboard

General purpose buckets

Directory buckets

General purpose buckets (2) Info

Buckets are containers for data stored in S3. Learn more

Find buckets by name

Copy ARN

Empty

Delete

Create bucket

	Name	AWS Region	Access	Creation date
<input type="radio"/>	elasticbeanstalk-us-east-2-891377370771	US East (Ohio) us-east-2	Objects can be public	January 30, 2024, 12:44:25 (UTC-08:00)
<input type="radio"/>	jaanu456	US East (Ohio) us-east-2	Bucket and objects not public	February 1, 2024, 09:22:46 (UTC-08:00)

```
aws Services Search [Alt+S] Ohio Jahnavi Yarlagaadda
ec2-user@ip-172-31-25-173 ~]$ ls
ec2-user@ip-172-31-25-173 ~]$ pwd
/home/ec2-user
ec2-user@ip-172-31-25-173 ~]$ touch jahnavi.sh
ec2-user@ip-172-31-25-173 ~]$ vi jahnavi.sh
ec2-user@ip-172-31-25-173 ~]$ ^C
ec2-user@ip-172-31-25-173 ~]$ cat jahnavi.sh
#!/bin/bash

SOURCE_DIR="/home/ec2-user/"
3_BUCKET="s3://jaanu456"

ws s3 sync $SOURCE_DIR $S3_BUCKET

ec2-user@ip-172-31-25-173 ~]$ vi jahnavi.sh
ec2-user@ip-172-31-25-173 ~]$ cat jahnavi.sh
#!/bin/bash

SOURCE_DIR="/home/ec2-user/"
3_BUCKET="s3://jaanu456"

ws s3 sync $SOURCE_DIR $S3_BUCKET

ec2-user@ip-172-31-25-173 ~]$ chmod +x jaanu456.sh
chmod: cannot access 'jaanu456.sh': No such file or directory
ec2-user@ip-172-31-25-173 ~]$ chmod +x jahnavi.sh
ec2-user@ip-172-31-25-173 ~]$

i-0081a8857c3f89169 (Jahnavi123)
PublicIPs: 3.23.113.156 PrivateIPs: 172.31.25.173
```

```
aws Services Search [Alt+S] Ohio Jahnavi Yarlagaadda
Running transaction
Preparing : 1/1
Installing : cronie-1.5.7-1.amzn2023.0.2.x86_64 1/2
Running scriptlet: cronie-1.5.7-1.amzn2023.0.2.x86_64 1/2
Created symlink /etc/systemd/system/multi-user.target.wants/crond.service -> /usr/lib/systemd/system/crond.service.

Installing : cronie-anacron-1.5.7-1.amzn2023.0.2.x86_64 2/2
Running scriptlet: cronie-anacron-1.5.7-1.amzn2023.0.2.x86_64 2/2
Verifying : cronie-anacron-1.5.7-1.amzn2023.0.2.x86_64 1/2
Verifying : cronie-1.5.7-1.amzn2023.0.2.x86_64 2/2

Installed:
cronie-1.5.7-1.amzn2023.0.2.x86_64 cronie-anacron-1.5.7-1.amzn2023.0.2.x86_64

Complete!
(ec2-user@ip-172-31-25-173 ~)$ crontab -e
no crontab for ec2-user - using an empty one
crontab: installing new crontab
(ec2-user@ip-172-31-25-173 ~)$ pwd
/home/ec2-user
(ec2-user@ip-172-31-25-173 ~)$ ^C
(ec2-user@ip-172-31-25-173 ~)$ ^C
(ec2-user@ip-172-31-25-173 ~)$ ^C
(ec2-user@ip-172-31-25-173 ~)$ crontab -e
crontab: installing new crontab
(ec2-user@ip-172-31-25-173 ~)$ chmod +x jahnavi.sh
(ec2-user@ip-172-31-25-173 ~)$

i-0081a8857c3f89169 (Jahnavi123)
PublicIPs: 3.23.113.156 PrivateIPs: 172.31.25.173
```

3) aws cli (aws s3 sync <source> <destination>)

Resource Groups & Tag Editor

Amazon S3

Buckets

Access Grants

Access Points

Object Lambda Access Points

Multi-Region Access Points

Batch Operations

IAM Access Analyzer for S3

Block Public Access settings for this account

Storage Lens

Dashboards

Storage Lens groups

AWS Organizations settings

Amazon S3

Buckets

janu123

Info

Objects

Properties

Permissions

Metrics

Management

Access Points

Objects (1) Info

Refresh

Copy S3 URI

Copy URL

Download

Open

Delete

Actions


Create folder

Upload

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

Find objects by prefix

< 1 > ⚙

<input type="checkbox"/>	Name	Type	Last modified	Size	Storage class
<input type="checkbox"/>	 vk.sh	sh	February 1, 2024, 10:17:39 (UTC-08:00)	179.0 B	Standard