

I AM ASSIGNMENT

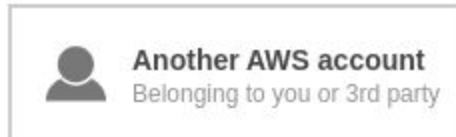
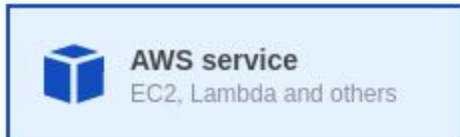


Name – Maithely Sharma
College – University of Petroleum and Energy Studies
EmployeeID – 4057

1. Create a Role with full access to S3

Create role

Select type of trusted entity



Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose a use case

Common use cases

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

Review

Provide the required information below and review this role before you create it.

Role name*

Maithelyrole

Use alphanumeric and '+=,._@-_' characters. Maximum 64 characters.

Role description

Allows EC2 instances to call AWS services on your behalf.

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

Trusted entities

AWS service: ec2.amazonaws.com

Policies

Policies not attached

Permissions boundary

Permissions boundary is not set

* Required

Cancel

Previous

Create role

Add permissions to Maithelyrole

Attach Permissions

Create policy



Filter policies ▾

Q s3

Showing 10 results

	Policy name ▾	Type	Used as
<input type="checkbox"/>	▶ AmazonDMSRedshiftS3Role	AWS managed	None
<input checked="" type="checkbox"/>	▶ AmazonS3FullAccess	AWS managed	Permissions policy (29)
<input type="checkbox"/>	▶ AmazonS3ReadOnlyAccess	AWS managed	Permissions policy (1)
<input type="checkbox"/>	▶ AWSLambdaS3ExecutionRole-1a3ddce4-a989-4828-88a4-7f5e701af3ef	Customer managed	Permissions policy (1)
<input type="checkbox"/>	▶ AWSLambdaS3ExecutionRole-34f3178e-e3ee-4238-8c64-0e27432978a2	Customer managed	Permissions policy (1)

Cancel

Attach policy

Summary

Policy ARN `arn:aws:iam::aws:policy/AmazonS3FullAccess` 

Description Provides full access to all buckets via the AWS Management Console.

Permissions | Policy usage | Policy versions | Access Advisor

Policy summary | { } JSON

Filter

Service	Access level	Resource	Request condition
Allow (1 of 223 services) Show remaining 222			
S3	Full access	All resources	None

2. Create another ROLE which has the policy to assume the previous Role

Create another role as Maithelyrole1 and attach policy of sts service to it

Service

STS

Actions

Manual actions

*

Resources

☒ Specific

☐ All resources

role

arn:aws:iam::187632318301:role/Maithelyrole

EDIT

*

☐ Any

[Add ARN to restrict access](#)

user

Any resource of type = user

☒ Any

Request conditions

[Specify request conditions \(optional\)](#)

Now attach this policy to the new role created

Attach policy

Attach the policy to users, groups, or roles in your account

Filter: Filter <input type="text" value="maithely"/>		Showing 4 results
<input type="checkbox"/> Name	Type	
<input type="checkbox"/> alice-maithely	User	
<input type="checkbox"/> maithely.sharma@tothenew.com	User	
<input type="checkbox"/> Maithelyrole	Role	
<input checked="" type="checkbox"/> MaithelyRole1	Role	

Now you can see that assume role attached

PermissionsTrust relationshipsTags (1)Access AdvisorRevoke sessions

▼ Permissions policies (2 policies applied)

Attach policies

+ Add inline policy

Policy name	Policy type	
▶ AmazonEC2FullAccess	AWS managed policy	✕
▶ assume-maithely-policy	Managed policy	✕

▶ Permissions boundary (not set)

Now create an ec2 instance and attach to the newrole created i.e Maithelyrole1

[Instances](#) > Attach/Replace IAM Role

Attach/Replace IAM Role

Select an IAM role to attach to your instance. If you don't have any IAM roles, choose Create new IAM role to create a role in the IAM console. If an IAM role is already attached to your instance, the IAM role you choose will replace the existing role.

Instance ID i-049890fc013ae0243 (ec2-maithely-java) ⓘ

IAM role*



[Create new IAM role](#)



* Required

Now add the arn of new role i.e maithelyrole1 to old role i.e maithelyrole in trust relationship

Edit Trust Relationship

You can customize trust relationships by editing the following access control policy document.

Policy Document

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Principal": {  
7         "AWS": "arn:aws:iam::187632318301:role/MaithelyRole1" ,|  
8       "Service": "ec2.amazonaws.com"  
9     },  
10    "Action": "sts:AssumeRole"  
11  }  
12 ]  
13 }
```

Now you have to ssh to the instance created and update it. Also install aws cli

```

maithely@maithely:~$ ssh -i "maithelykeypair.pem" ubuntu@ec2-52-207-215-48.compute-1.amazonaws.com
Warning: Identity file maithelykeypair.pem not accessible: No such file or directory.
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri Feb 28 10:51:26 UTC 2020

System load:  0.08               Processes:            86
Usage of /:   16.5% of 7.69GB    Users logged in:     0
Memory usage: 17%               IP address for eth0: 172.31.104.188
Swap usage:   0%

54 packages can be updated.
32 updates are security updates.

Last login: Fri Feb 28 06:46:24 2020 from 182.71.160.186
ubuntu@ip-172-31-104-188:~$ sudo apt update
Hit:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic InRelease
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates InRelease [88.7 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-backports InRelease [74.6 kB]
Get:4 http://security.ubuntu.com/ubuntu bionic-security InRelease [88.7 kB]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages [871 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu bionic-updates/universe amd64 Packages [1054 kB]
Fetched 2177 kB in 1s (2789 kB/s)
Reading package lists... Done
Building dependency tree
Reading state information... Done
51 packages can be upgraded. Run 'apt list --upgradable' to see them.
ubuntu@ip-172-31-104-188:~$ sudo apt install awscli
Reading package lists... Done
Building dependency tree

```

3. Attach this to an instance and get an sts token.

Now copy the arn of s3fullaccess i.e of role maithelyrole

```

ubuntu@ip-172-31-104-188:~$ aws sts assume-role --role-arn arn:aws:iam::187632318301:role/Maithelyrole --role-session-name maithelyrolefisrt
{
  "Credentials": {
    "AccessKeyId": "ASIASXL6B650XC2SPQK6",
    "SecretAccessKey": "P2jVsE5F55CoQRr+A6gfx8gcc5lVykrLVeC1XNuz",
    "SessionToken": "FwoGZXIvYXZlEBwaDAYWUBAJ8D8sJISLFCK1AY+NiEmwjx+/cw7N73N2CqFl++6z9k7UT42yiv83xAjmi9ARfBlibv066PFgL4IX9IJW02Y82QDbm4DxrmvL5qedjSXh0pn2jjDhSWWqWy+8tkxGLGPxnzhkGoKJNTAXpSujT5aVrag5DKpAmat5o8Mceg2cCgl1QcTZUeGcu7ZUaToIJRxnkk9Yrf0GVAHePvw44V6G9VjqWIJmGelmRm5+lsAS21bsPinCcKZ0P0kFL8m+MdgoiOPj8gUyLUirtBybpXdB9SLycNahYCj/8mE2JtFAHfzPkcHfdTH7YFqN0vjhkaVTJbRqRQ==",
    "Expiration": "2020-02-28T11:55:04Z"
  },
  "AssumedRoleUser": {
    "AssumedRoleId": "AROASXL6B650RPZY5NT2B:maithelyrolefisrt",
    "Arn": "arn:aws:sts::187632318301:assumed-role/Maithelyrole/maithelyrolefisrt"
  }
}

```


Now export it:

```
ubuntu@ip-172-31-104-188:~$ export AWS_ACCESS_KEY_ID=ASIASXL6B650XC2SPQK6
ubuntu@ip-172-31-104-188:~$ export AWS_SECRET_ACCESS_KEY=P2jVsE5F55CoQRr+A6gfx8g
cc5lVykrLVeC1XNuz
ubuntu@ip-172-31-104-188:~$
ubuntu@ip-172-31-104-188:~$ export AWS_SESSION_TOKEN=FwoGZXIvYXdzEBwaDAYWUBAJ8D
8sJISLFCK1AY+NiEmwjx+/cw7N73N2CqFl++6z9k7UT42yiv83xAjmi9ARfBlibv066PFgL4IX9IJW02
Y82QDbm4DxrmvL5qedjSXh0pn2jjDhSWWqWy+8tkxGLGPxnzhkGoKJNtAXpSujT5aVrag5DKpAmat5o8
Mceg2cCgl1QcTZUeGcu7ZUaToIJRxnkk9Yrf0GVAHePvw44V6G9VjqWlJmGelmRm5+lsAS21bsPinCcK
Z0P0kFL8m+MdgoiOPj8gUyLUirtBybpXdB9SLycNahYCj/8mE2JtFAHfzPkcHfdTH7YFqN0vjhkaVTJb
RqRQ==
```

Now you can access s3

```
ubuntu@ip-172-31-104-188:~$ aws s3 ls
2019-06-26 12:11:08 0testuser11
2018-04-20 16:59:22 187632318301-awsmacietrail-dataevent
2019-04-02 10:11:33 7testdemo
2019-03-11 04:51:59 abhimanyucftemplate
2020-02-28 10:55:02 abhishek-bootcamp
2019-03-04 06:55:23 abneesh1
2019-03-11 11:00:41 adityamun007
2020-02-26 16:26:29 akshaybuck1
2020-02-27 08:55:25 aman-khandelwal-1
2019-03-07 09:40:48 anmol-bootcamp19
2019-03-08 00:25:58 avcabc
2017-09-07 03:41:42 aws-codestar-us-east-1-187632318301
2017-09-07 04:23:01 aws-codestar-us-east-1-187632318301-codestartest2-app
2017-09-07 04:23:07 aws-codestar-us-east-1-187632318301-codestartest2-pipe
2017-09-07 03:41:48 aws-codestar-us-east-1-187632318301-codestarttest-pipe
2019-06-26 05:39:55 aws-lambda-trigger-ronozor
2020-02-28 03:56:49 ayush-public-bucket
2020-02-25 07:02:11 baban-123
2018-02-14 12:28:43 cf-templates-71mx96oilvv5-us-east-1
```

4. Create a group for "Data Administrator" where the user 'Alice' be a member of this group. This group will prepare the data for the analysis. So Provide the following access to the group.

Service: Amazon S3;

Action:

Get*,

List*,

Put*,

ARN: Input and output Buckets (no conditions)

Firstly create a user alice-maithely

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name*

[+ Add another user](#)

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

- Access type* ☐ **Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.
- ☒ **AWS Management Console access**
Enables a **password** that allows users to sign-in to the AWS Management Console.

- Console password* ☐ Autogenerated password
☒ Custom password

* Required

[Cancel](#)

[Next: Permissions](#)

Add the specified services to it

Service

S3

Actions

close

Specify the actions allowed in S3 ?

Switch to deny permissions ?

Manual actions (add actions)

☐ All S3 actions (s3:*)

Access level

☒ List (3 selected)

☒ Read (41 selected)

☐ Tagging

☐ DeleteObjectTagging ?

☐ PutBucketTagging ?

☐ PutObjectVersionTagging ?

☐ DeleteObjectVersionTagging ?

☐ PutObjectTagging ?

☐ ReplicateTags ?

☒ Write (31 selected)

☐ Permissions management

Expand all | Collapse all

Resources

close

☒ Specific
 ☐ All resources

accesspoint ?	Any resource of type = accesspoint	<input checked="" type="checkbox"/> Any
bucket ?	Any resource of type = bucket	<input checked="" type="checkbox"/> Any
job ?	Any resource of type = job	<input checked="" type="checkbox"/> Any
object ?	Any resource of type = object	<input checked="" type="checkbox"/> Any

▶ Request conditions
 [Specify request conditions \(optional\)](#)

[Add additional permissions](#)

Character count: 2,197 of 6,144.

[Cancel](#)

[Review policy](#)

Now create a policy

Review policy

Name*

alice-s3-maithely

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

Description

read , write and list permissions

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

Summary

Q Filter

Service ▾	Access level	Resource	Request condition
Allow (1 of 223 services) Show remaining 222			
S3	Full: List, Read, Write	Multiple	None

Now attach this policy to the group created of data administrator

Attach policy

Attach the policy to users, groups, or roles in your account

Filter: Filter ▾		Q Data		Showing 8 results	
<input type="checkbox"/>	Name ▾			Type ▾	
<input type="checkbox"/>	data-administrator			Group	
<input type="checkbox"/>	DataAdmin			Group	
<input type="checkbox"/>	DataAdmin-Chhavi			Group	
<input checked="" type="checkbox"/>	DataAdministrator			Group	

5. Create a group for the "Developer group " where the user 'bob ' is a member of this group. This group with Test Newly Developed Features for which they require access to EC2 instances. Provide the following access to this group:

Service: Amazon EC2

Action: *Instances, *Volume, Describe*, CreateTags;

Condition: Dev Subnets only

Firstly create a group "Developer-Maithely" and grant EC2fullaccess to it

Set Group Name

Specify a group name. Group names can be edited any time.

Group Name:

Example: Developers or ProjectAlpha

Maximum 128 characters

Attach Policy

Select one or more policies to attach. Each group can have up to 10 policies attached.

Filter: Policy Type <input type="text" value="Search"/>		Showing 583 results		
	Policy Name ↕	Attached Entities ↕	Creation Time ↕	
<input type="checkbox"/>	 AmazonS3FullAccess	30	2015-02-07 00:10 UTC...	
<input type="checkbox"/>	 AmazonEC2ContainerServiceforEC2Role	17	2015-03-20 00:15 UTC...	
<input type="checkbox"/>	 AmazonEC2ContainerRegistryFullAccess	16	2015-12-21 22:36 UTC...	
<input checked="" type="checkbox"/>	 AmazonEC2FullAccess	16	2015-02-07 00:10 UTC...	
<input type="checkbox"/>	 CloudWatchFullAccess	8	2015-02-07 00:10 UTC...	
<input type="checkbox"/>	 AmazonSNSFullAccess	6	2015-02-07 00:11 UTC...	
<input type="checkbox"/>	 AmazonDynamoDBFullAccess	5	2015-02-07 00:10 UTC...	
<input type="checkbox"/>	 CloudWatchLogsFullAccess	2	2015-02-07 00:10 UTC...	
<div>Cancel Previous Next Step</div>				

Now create a BOB user

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name*

[+ Add another user](#)

Select AWS access type

Select how these users will access AWS. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Access type* ☐ **Programmatic access**
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

☒ **AWS Management Console access**
Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password* ☐ Autogenerated password
☒ Custom password

* Required

Cancel

Next: Permissions

Add the user:BOB to the “Developers” group

Add user to group

Create groupRefresh

Showing 1 result

Group	Attached policies
<input checked="" type="checkbox"/> Developer-Maithely	AmazonEC2FullAccess

Now create a subnet which is going to get attached to the policy

Create subnet

Enter your subnet's IP address block in CIDR format; for example, 10.0.0.0/24. IPv4 block sizes must be between a /16 netmask and /28 netmask. IPv6 block must be a /64 CIDR block.

Name tag

maithely-subnet

VPC*

vpc-00470a42fc196d84e

Availability Zone

us-east-1c

The Availability Zone where this subnet will reside. Select No Preference to let Amazon choose an Availability Zone for you.

VPC CIDRs

CIDR	Status	Subnet
10.0.0.0/16	associated	

IPv4 CIDR block*

10.0.0.0/24

Now create a policy

Expand all | Collapse all

EC2 (5 actions)

Clone | Remove

Service

EC2

Actions

List

DescribeInstances

DescribeVolumes

Read

DescribeTags

Tagging

CreateTags

DeleteTags

Resources

Specific

close

All resources

canacitiv-reservati ⓘ Any resource of type = canacitiv-reservation ✓ Any

In that policy you have to add ARN for that specified subnet

console.aws.amazon.com/iam/home?#/policies\$new?step=edit

Services ▾ Resource C

@tothenew.com ... ▾ Global ▾ Support ▾

Add ARN(s)

×

Amazon Resource Names (ARNs) uniquely identify AWS resources. Resources are unique to each service. [Learn more](#)

Specify ARN for subnet [List ARNs manually](#)

arn:aws:ec2:us-east-1:187632318301:subnet/subnet-04f827cafa2fd11d4

Region *

us-east-1

Any

Account *

187632318301

Any

Subnet id *

subnet-04f827cafa2fd11d4

Any

Cancel

Add

traffic-mirror-filter ⓘ Any resource of type = traffic-mirror-filter ✓ Any

traffic-mirror-sess... ⓘ Any resource of type = traffic-mirror-session ✓ Any

After the policy has been created, you have to attach this policy to the Developers group

Attach policy

Attach the policy to users, groups, or roles in your account

Filter: Filter Q Deve		Showing 4 results
<input type="checkbox"/> Name ▼	Type ▼	
<input type="checkbox"/> lambda-developer-identity-provider-role	Role	
<input type="checkbox"/> developer-group-ayush	Group	
<input checked="" type="checkbox"/> Developer-Maithely	Group	
<input type="checkbox"/> Developers	Group	

Now you can see that in Developers group that the policy has been attached

▼ Summary


Group ARN:	arn:aws:iam::187632318301:group/Developer-Maithely 
Users (in this group):	1
Path:	/
Creation Time:	2020-02-27 23:10 UTC+0530

Users **Permissions** **Access Advisor**

Managed Policies

The following managed policies are attached to this group. You can attach up to 10 managed policies.

[Attach Policy](#)

Policy Name	Actions
 AmazonEC2FullAccess	Show Policy Detach Policy Simulate Policy
maithely-policy	Show Policy Detach Policy Simulate Policy

6. Identify the unused IAM users/credentials using AWS CLI.

For seeing the unused password through AWS cli:

Lists the IAM users that have the specified path prefix. If no path prefix is specified, the operation returns all users in the AWS account. If there are none, the operation returns an empty list.

```
maithely@maithely:~$ aws iam list-users
{
  "Users": [
    {
      "Path": "/",
      "UserName": "abhishek.chauhan1@tothenew.com",
      "UserId": "AIDASXL6B650Q4RMZ427Z",
      "Arn": "arn:aws:iam::187632318301:user/abhishek.chauhan1@tothenew.com",
      "CreateDate": "2020-02-19T11:03:23Z",
      "PasswordLastUsed": "2020-02-28T05:03:08Z"
    },
    {
      "Path": "/",
      "UserName": "aditya.upadhyay@tothenew.com",
      "UserId": "AIDASXL6B650YD7UUCZUJ",
      "Arn": "arn:aws:iam::187632318301:user/aditya.upadhyay@tothenew.com",
      "CreateDate": "2020-02-19T11:03:25Z",
      "PasswordLastUsed": "2020-02-28T04:46:17Z"
    },
    {
      "Path": "/",

```

```
{
  "Path": "/",
  "UserName": "Gargi_Alice",
  "UserId": "AIDASXL6B650WNUVPT664",
  "Arn": "arn:aws:iam::187632318301:user/Gargi_Alice",
  "CreateDate": "2020-02-27T10:45:35Z"
},
{
  "Path": "/",
  "UserName": "garima.dabral@tothenew.com",
  "UserId": "AIDASXL6B650ZMUKAFYLP",
  "Arn": "arn:aws:iam::187632318301:user/garima.dabral@tothenew.com",
  "CreateDate": "2020-02-19T11:03:44Z"
},
{
  "Path": "/"

```

```
maithely@maithely:~$ aws iam list-users |jq '.Users[] | select(.PasswordLastUsed==null) |.UserName '
"Alice"
"Alice-Chhavi"
"alice-maithely"
"asusumeuser"
"Bob"
"Bob-maithely"
"bobpooja"
"CloudCheckr"
"dikshaTomar"
"Gargi_Alice"
"garima.dabral@tothenew.com"
"HAWK2.0-user"
"poojaalice"
"raghu.sharma@tothenew.com"
"s3pooja"
"vivek.yadavi@tothenew.com"
maithely@maithely:~$
```

For seeing unused access keys:

Returns information about the access key IDs associated with the specified IAM user. If there is none, the operation returns an empty list.

```
maithely@maithely:~$ aws iam list-access-keys
{
  "AccessKeyMetadata": [
    {
      "UserName": "maithely.sharma@tothenew.com",
      "AccessKeyId": "AKIASXL6B650XQFOJAKS",
      "Status": "Active",
      "CreateDate": "2020-02-27T10:35:52Z"
    }
  ]
}
```

```
maithely@maithely:~$ aws iam get-access-key-last-used --access-key-id AKIASXL6B650XQFOJAKS
{
  "UserName": "maithely.sharma@tothenew.com",
  "AccessKeyLastUsed": {
    "LastUsedDate": "2020-02-28T05:38:00Z",
    "ServiceName": "iam",
    "Region": "us-east-1"
  }
}
```

7. Identify all the instances having the tag key-value "backup=true" using AWS CLI.

```
maithely@maithely:~$ aws ec2 describe-instances --filters "Name=tag:backup,Values=true"
{
  "Reservations": []
}
```


8. An EC2 Instance hosts a Java-based application that accesses an s3 bucket. This EC2 Instance is currently serving production users. Create the role and assign the role to EC2 instance.


Create a role with ec2 service


Create role


1 2 3 4

Select type of trusted entity

**AWS service**
EC2, Lambda and others

**Another AWS account**
Belonging to you or 3rd party

**Web identity**
Cognito or any OpenID provider

**SAML 2.0 federation**
Your corporate directory

Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose a use case

Common use cases

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

Now give S3 permission

Choose one or more policies to attach to your new role.

Create policy↺

Filter policies ▾

Q s3

Showing 16 results

	Policy name ▾	Used as
<input type="checkbox"/>	alice-s3-maithely	Permissions policy (2)
<input type="checkbox"/>	AmazonDMSRedshiftS3Role	None
<input checked="" type="checkbox"/>	AmazonS3FullAccess	Permissions policy (35)
<input type="checkbox"/>	AmazonS3ReadOnlyAccess	Permissions policy (1)
<input type="checkbox"/>	AWSLambdaExecutionRole-1a7ddac4-c000-4000-80e4-75e701e02d	Permissions policy (1)

Now create an instance and attach it to role

Attach/Replace IAM Role

Select an IAM role to attach to your instance. If you don't have any IAM roles, choose Create new IAM role to create a role in the IAM console. If an IAM role is already attached to your instance, the IAM role you choose will replace the existing role.

Instance ID i-049890fc013ae0243 (ec2-maithely-java) ⓘ

IAM role* ↺ [Create new IAM role](#) ⓘ

* Required

Now to check this


```
maithely@maithely:~$ ssh -i "maithelykeypair.pem" ubuntu@ec2-52-207-215-48.compute-1.amazonaws.com
Warning: Identity file maithelykeypair.pem not accessible: No such file or directory.
The authenticity of host 'ec2-52-207-215-48.compute-1.amazonaws.com (52.207.215.48)' can't be established.
ECDSA key fingerprint is SHA256:GumXvaCiY/WqkXUc+g2KxYBZ7eMyvjPiIXto98k39P0.
Are you sure you want to continue connecting (yes/no)? YES
Warning: Permanently added 'ec2-52-207-215-48.compute-1.amazonaws.com,52.207.215.48' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 18.04.3 LTS (GNU/Linux 4.15.0-1057-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

System information as of Fri Feb 28 06:46:23 UTC 2020

System load:  0.0               Processes:            88
Usage of /:   13.6% of 7.69GB   Users logged in:     0
Memory usage: 15%              IP address for eth0: 172.31.104.188
Swap usage:   0%

0 packages can be updated.
0 updates are security updates.
```

```

ubuntu@ip-172-31-104-188:~$ aws s3 ls
2019-06-26 12:11:08 0testuser11
2018-04-20 16:59:22 187632318301-awsmacietrail-dataevent
2019-04-02 10:11:33 7testdemo
2019-03-11 04:51:59 abhimanyucftemplate
2019-03-04 06:55:23 abneesh1
2019-03-11 11:00:41 adityamun007
2020-02-26 16:26:29 akshaybuck1
2020-02-27 08:55:25 aman-khandelwal-1
2019-03-07 09:40:48 anmol-bootcamp19
2019-03-08 00:25:58 avcabc
2017-09-07 03:41:42 aws-codestar-us-east-1-187632318301
2017-09-07 04:23:01 aws-codestar-us-east-1-187632318301-codestartest2-app
2017-09-07 04:23:07 aws-codestar-us-east-1-187632318301-codestartest2-pipe
2017-09-07 03:41:48 aws-codestar-us-east-1-187632318301-codestarttest-pipe
2019-06-26 05:39:55 aws-lambda-trigger-ronozor
2020-02-28 03:56:49 ayush-public-bucket
2020-02-25 07:02:11 baban-123
2018-02-14 12:28:43 cf-templates-71mx96ojlvv5-us-east-1
2019-03-27 15:57:27 cfront1
2020-02-26 11:51:54 chirag-bucket-2
2020-02-26 11:46:43 chirag-bucket1
2019-03-27 20:34:52 cloudfront8
2020-02-25 10:59:18 copy-test-delete
2020-02-26 08:17:11 diksha.static.website
2019-06-26 10:49:10 ec2-access-bucket
2019-03-28 05:23:51 ec2-ttn
2019-03-01 07:28:00 ekanshbucket
2019-03-14 10:29:37 elasticbeanstalk-us-east-1-187632318301
2016-10-17 07:46:10 elasticbeanstalk-us-west-2-187632318301
2017-10-06 09:11:17 geekcombat-ttn
2020-02-27 15:22:31 kaushubucket
2019-06-25 05:48:05 mynewtrialbucket
2020-02-27 04:58:00 :

```

9. You have both production and development based instances running on your VPC. It is required to ensure that people responsible for the development instances do not have access to work on production instances for better security. Define the tags on the test and production servers and add a condition to the IAMPolicy which allows access to specific tags.

We have create two instances in the default VPC:

- 1)maithely-production
- 2)maithely-development

Launch Instance Connect Actions

search : maithely Add filter

	Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public
<input checked="" type="checkbox"/>	maithely-development	i-024b49adf39d09059	t2.micro	us-east-1c	running	Initializing	None	ec2-3-
<input type="checkbox"/>	ec2-maithely-java	i-049890fc013ae0243	t2.micro	us-east-1c	running	2/2 checks ...	None	ec2-52
<input checked="" type="checkbox"/>	maithely-production	i-09e07adc50ef79b22	t2.micro	us-east-1c	running	Initializing	None	ec2-3-

Now create 2 users :

1)Dev1-maithely

2)prod1-maithely

Add user Delete user

search : maithely Showing 5

	User name	Groups	Access key age	Password age	Last activity	MFA
<input type="checkbox"/>	alice-maithely	DataAdministrator	None	Today	None	Not ena
<input type="checkbox"/>	Bob-maithely	Developer-Maithely	None	Today	None	Not ena
<input checked="" type="checkbox"/>	Dev1-maithely	None	None	Today	None	Not ena
<input type="checkbox"/>	maithely.sh...	BootCamp2019	Today	8 days	Today	Not ena
<input checked="" type="checkbox"/>	prod1-maith...	None	None	Today	None	Not ena

Now create a policy for development server

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor JSON Import managed policy

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "VisualEditor0",
6       "Effect": "Allow",
7       "Action": [
8         "ec2:StartInstances",
9         "ec2:StopInstances",
10        "ec2:DescribeInstances"
11      ]
12    }
13  ]
14 }

```

Visual editorJSONImport managed policy

```

11      ],
12      "Resource": "arn:aws:ec2:us-east-1:i-024b49adf39d09059:instance/*",
13      "Condition": {
14        "StringEquals": {
15          "ec2:ResourceTag/Name": "maithelydevelopment",
16          "aws:PrincipalTag/Name": "Dev1-maithely"
17        }
18      }
19    }
20  ]
21 }

```

And similarly for production server

10. Create a policy for allowing users to set or rotate their credentials, such as their console password, their programmatic access keys, and their MFA devices.

Create a policy that has all these actions

Service

IAM

Actions

List

ListAccessKeys

ListMFADevices

ListPoliciesGrantingServiceAccess

ListVirtualMFADevices

Read

GenerateOrganizationsAccessReport

GenerateServiceLastAccessedDetails

GetAccessKeyLastUsed

GetAccountPasswordPolicy

GetOrganizationsAccessReport

GetServiceLastAccessedDetails

GetServiceLastAccessedDetailsWithEntities

Write

ChangePassword

CreateAccessKey

CreateVirtualMFADevice

DeactivateMFADevice

DeleteAccessKey

DeleteVirtualMFADevice

EnableMFADevice

PassRole

ResyncMFADevice

UpdateAccessKey

UpdateAccountPasswordPolicy

Permissions management

DeleteAccountPasswordPolicy

Resources

Specific

All resources