

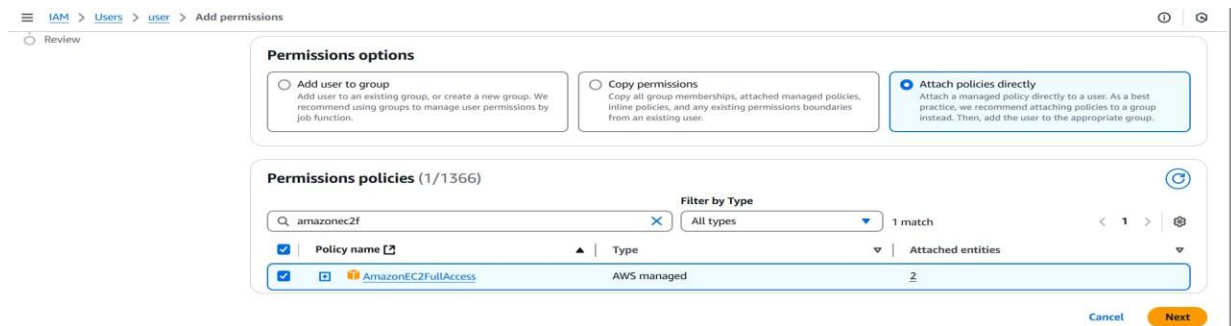
**Task 3 :** An organisation has multiple ec2 instances running in a VPC on AWS. Each instance has multiple tags like “Department” and “Hostname” associated with it. The value of the Department tag is one of Frontend, Backend, DevOps, QA, ITOps, etc. The organisation has requested you to change the Department tags of a few servers and have shared a csv file with the old and new values for Hotsname.

Write a python or Java Code for modifying the value of the Department tag by reading the csv file and accordingly changing the values of the following instances.

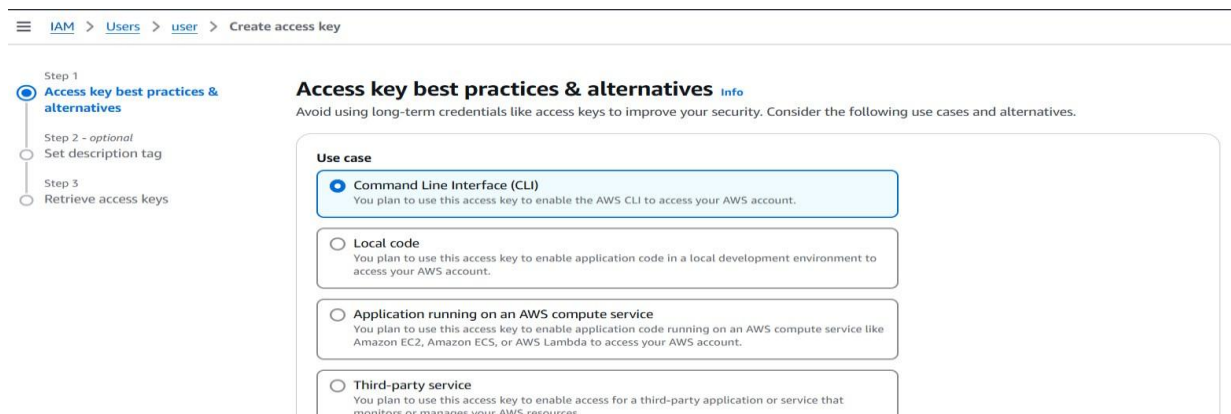
AWS CLI is installed and configured:

open the IAM and configure the process User->Create User->give the user name.

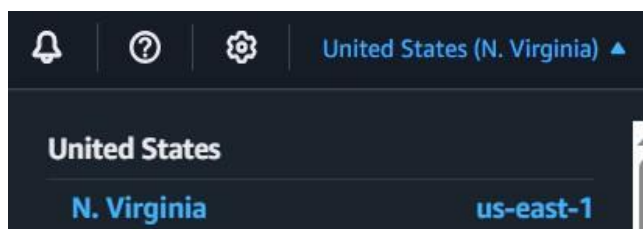
Give full access to the ec2 user.->create user.



Open the created user and give the access key->Security Credentials->CLI->Create access key.



My Region :



commands:

sudo snap install aws-cli -- classic

```
control@control:~$ snap install aws-cli --classic
error: access denied (try with sudo)
control@control:~$ sudo snap install aws-cli --classic
[sudo] password for control:
aws-cli (v2/stable) 2.26.6 from Amazon Web Services (aws✓) installed
control@control:~$
```

Then Configure the AWS with the command : aws configure

```
control@control:~$ aws configure
AWS Access Key ID [None]: AKIA3C6FMEXR5MAKCVME
AWS Secret Access Key [None]: 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
Default region name [None]: us-east-1
Default output format [None]:
control@control:~$
```

Update the CSV file .

```
control@control:~$ cat > tag_update.csv <<EOF
> Hostname,Current Department,New Department
> webserver01,Backend,Frontend
> webserver02,Frontend,Frontend
> database04,QA,Backend
> ansible,Devops,Backend
> scanner,,Devops
> activedirectory02,ITops,ITops
> database03,Frontend,Backend
> jenkins,,Devops
> EOF
```

Install the python and Boto3 library.

commands:

**Sudo apt python3-pip**

**pip install boto3 pandas**

**Sudo python3 -m venv boto -env**

**python3 -m venv boto -env**

```
(boto-env) control@control:~$ pip install boto3 pandas
Collecting boto3
  Downloading boto3-1.37.38-py3-none-any.whl.metadata (6.7 kB)
Collecting pandas
  Downloading pandas-2.2.3-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (89 kB)
----- 89.9/89.9 kB 3.6 MB/s eta 0:00:00
Collecting botocore<1.38.0,>=1.37.38 (from boto3)
  Downloading botocore-1.37.38-py3-none-any.whl.metadata (5.7 kB)
Collecting jmespath<2.0.0,>=0.7.1 (from boto3)
  Downloading jmespath-1.0.1-py3-none-any.whl.metadata (7.6 kB)
Collecting s3transfer<0.12.0,>=0.11.0 (from boto3)
  Downloading s3transfer-0.11.5-py3-none-any.whl.metadata (1.7 kB)
Collecting numpy>=1.26.0 (from pandas)
  Downloading numpy-2.2.5-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (62 kB)
----- 62.0/62.0 kB 4.9 MB/s eta 0:00:00
Collecting python-dateutil>=2.8.2 (from pandas)
  Downloading python_dateutil-2.9.0.post0-py2.py3-none-any.whl.metadata (8.4 kB)
Collecting pytz>=2020.1 (from pandas)
  Downloading pytz-2025.2-py2.py3-none-any.whl.metadata (22 kB)
Collecting tzdata>=2022.7 (from pandas)
  Downloading tzdata-2025.2-py2.py3-none-any.whl.metadata (1.4 kB)
Collecting urllib3<2.2.0,>=1.25.4 (from botocore<1.38.0,>=1.37.38->boto3)
  Downloading urllib3-2.4.0-py3-none-any.whl.metadata (6.5 kB)
Collecting six>=1.5 (from python-dateutil>=2.8.2->pandas)
  Downloading six-1.17.0-py2.py3-none-any.whl.metadata (1.7 kB)
Downloaded boto3-1.37.38-py3-none-any.whl (139 kB)
----- 139.9/139.9 kB 8.1 MB/s eta 0:00:00
Downloaded pandas-2.2.3-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (12.7 MB)
----- 12.7/12.7 MB 11.1 MB/s eta 0:00:00
Downloaded botocore-1.37.38-py3-none-any.whl (13.5 MB)
----- 13.5/13.5 MB 11.0 MB/s eta 0:00:00
Downloaded jmespath-1.0.1-py3-none-any.whl (20 kB)
----- 13.5/13.5 MB 11.0 MB/s eta 0:00:00
Downloaded numpy-2.2.5-cp312-cp312-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (16.1 MB)
----- 16.1/16.1 MB 11.8 MB/s eta 0:00:00
Downloaded python_dateutil-2.9.0.post0-py2.py3-none-any.whl (229 kB)
----- 229.9/229.9 kB 6.7 MB/s eta 0:00:00
Downloaded pytz-2025.2-py2.py3-none-any.whl (509 kB)
----- 509.2/509.2 kB 8.4 MB/s eta 0:00:00
Downloaded s3transfer-0.11.5-py3-none-any.whl (84 kB)
----- 84.0/84.0 kB 5.0 MB/s eta 0:00:00
Downloaded tzdata-2025.2-py2.py3-none-any.whl (347 kB)
----- 347.8/347.8 kB 18.9 MB/s eta 0:00:00
Downloaded six-1.17.0-py2.py3-none-any.whl (11 kB)
----- 11.0/11.0 kB 11.0 MB/s eta 0:00:00
Downloaded urllib3-2.4.0-py3-none-any.whl (128 kB)
----- 128.7/128.7 kB 6.6 MB/s eta 0:00:00
Installing collected packages: pytz, urllib3, tzdata, six, numpy, jmespath, python-dateutil, pandas, botocore, s3transfer, boto3
```

For Creating the instance:

commands:

```
aws ec2 run-instances \  
  --image-id ami-0c02fb55956c7d316 \  
  --instance-type t2.micro \  
  --key-name my-key \  
  --security-groups my-security-group \  
  --region us-east-1
```

TO RUN THE INSTANCE WITH THE TAG :

Then the department will correctly.

```
aws ec2 run-instances \  
  --image-id ami-0c02fb55956c7d316 \  
  --instance-type t2.micro \  
  --key-name my-key \  
  --security-groups default \  
  --tag-specifications  
'ResourceType=instance,Tags=[{Key=Hostname,Value=database04},{Key=Department,Value=  
QA1}]' \  
  --count 1
```

TO RUN THE COMMANDS FOR THE OUTPUT TABLE:

```
aws ec2 describe-instances --query "Reservations[*].Instances[*].[InstanceId,  
Tags[?Key=='Hostname'].Value | [0], Tags[?Key=='Department'].Value | [0]]" --output table
```

```
(boto-env) control@control:~/abc$ aws ec2 describe-instances --query "Reservations[*].Instances[*].[InstanceId, Tags[?Key=='Hostname'].Value | [0], Tags[?Key=='Department'].Value | [0]]" --  
output table  
  
-----  
| DescribeInstances |  
+-----+-----+  
| i-0e584173cbc4ec79c | database04 | Backend |  
| i-0a69e5fd1186fda53 | database04 | QA |  
| i-008464366b5f51995 | database04 | QA |  
| i-0b34fa8f05cfe1450 | webserver01 | Frontend |  
| i-06085794d8887a141 | webserver02 | Frontend |  
| i-09df205103930cedb | database04 | Backend |  
+-----+-----+-----+
```

Instances (4) [Info](#) Last updated less than a minute ago [Connect](#) [Instance state](#) [Actions](#) [Launch instances](#)

Find Instance by attribute or tag (case-sensitive) [All states](#)

Instance state = running [Clear filters](#) < 1 > [Settings](#)

<input type="checkbox"/>	Name <a href="#">↗</a>	Instance ID	Instance state <a href="#">▼</a>	Instance type <a href="#">▼</a>	Status check	Alarm status	Availability Zone <a href="#">▼</a>	Public IP
<input type="checkbox"/>		i-0e584173cbc4ec79c	Running <a href="#">🔍</a> <a href="#">🔍</a>	t2.micro	2/2 checks passed	<a href="#">View alarms +</a>	us-east-1c	ec2-54-16
<input type="checkbox"/>	<a href="#">🔗</a>	i-0b34fa8f05cfe1450	Running <a href="#">🔍</a> <a href="#">🔍</a>	m1.small	2/2 checks passed	<a href="#">View alarms +</a>	us-east-1c	ec2-44-2
<input type="checkbox"/>		i-06085754d8887a141	Running <a href="#">🔍</a> <a href="#">🔍</a>	t2.micro	2/2 checks passed	<a href="#">View alarms +</a>	us-east-1c	ec2-3-88-
<input type="checkbox"/>		i-09df205103930cedb	Running <a href="#">🔍</a> <a href="#">🔍</a>	t2.micro	2/2 checks passed	<a href="#">View alarms +</a>	us-east-1c	ec2-3-86-

## update python.py

update python.py script is used to see the AWS Console and check the tags of those EC2 instances manually.

command:

`aws ec2 describe-instances --region us-east-1 --query "Reservations[*].Instances[*].[InstanceId,Tags]"`

```
[{"InstanceId": "i-0e584173cbc4ec79c",
  "Tags": [
    {
      "Key": "Hostname",
      "Value": "database04"
    },
    {
      "Key": "Department",
      "Value": "Backend"
    }
  ]
},
{"InstanceId": "i-0a69e5fd1186fda53",
  "Tags": [
    {
      "Key": "Hostname",
      "Value": "database04"
    },
    {
      "Key": "Department",
      "Value": "QA"
    }
  ]
},
{"InstanceId": "i-008464366b5f51995",
  "Tags": [
    {
      "Key": "Hostname",
      "Value": "database04"
    },
    {
      "Key": "Department",
      "Value": "QA1"
    }
  ]
},
{"InstanceId": "i-0b34fa8f05cfe1450",
  "Tags": [
```

Run this command to see the tags for the specific id value.

command:

**aws ec2 describe-tags --filters "Name=resource-id,Values=i-06085754d8887a141"**

```
(boto-env) control@control:~/abc$ vim tag_update.csv
(boto-env) control@control:~/abc$ aws ec2 describe-tags --filters "Name=resource-id,Values=i-0b34fa8f05cfe1450"
{
  "Tags": [
    {
      "Key": "Department",
      "ResourceId": "i-0b34fa8f05cfe1450",
      "ResourceType": "instance",
      "Value": "Frontend"
    },
    {
      "Key": "Hostname",
      "ResourceId": "i-0b34fa8f05cfe1450",
      "ResourceType": "instance",
      "Value": "webserver01"
    }
  ]
}
```

python de.py

Loading the CSV correctly. Using filters to identify the EC2 instances by Hostname and current Department.Updating the Department tag if the match is found.

```
(boto-env) control@control:~/abc$ python de.py
CSV Data Loaded:
  Hostname Current Department New Department
0 webserver01 Backend Frontend
1 webserver02 Frontend Frontend
2 database04 QA Backend
3 ansible Devops Backend
4 scanner NaN Devops
5 activedirectory02 ITops ITops
6 database03 Frontend Backend
7 jenkins NaN Devops
Processing instance webserver01 with current department: Backend and new department: Frontend
No matching instance found for webserver01 with Department Backend
Processing instance webserver02 with current department: Frontend and new department: Frontend
Found instance i-06085754d8887a141. Updating department tag...
Updated instance i-06085754d8887a141 with new Department: Frontend
Processing instance database04 with current department: QA and new department: Backend
Found instance i-0a69e5fd1186fda53. Updating department tag...
Updated instance i-0a69e5fd1186fda53 with new Department: Backend
Processing instance ansible with current department: Devops and new department: Backend
No matching instance found for ansible with Department Devops
Skipping scanner because it has no current department value.
Processing instance activedirectory02 with current department: ITops and new department: ITops
No matching instance found for activedirectory02 with Department ITops
Processing instance database03 with current department: Frontend and new department: Backend
No matching instance found for database03 with Department Frontend
Skipping jenkins because it has no current department value.
```

**TASK 3 OUTPUT:**

Modifying the value of the Department tag by reading the csv file and accordingly changing the values.