D: 07/07/2020

Windows Open Browser : http://169.254.169.254/latest/meta-data/

Linux : curl http://169.254.169.254/latest/meta-data/

curl http://169.254.169.254/latest/user-data/

ami-id

ami-launch-index

ami-manifest-path

block-device-mapping/

events/

hibernation/

hostname

identity-credentials/

instance-action

instance-id

instance-type

local-hostname

local-ipv4

mac

metrics/

network/

placement/

profile

public-hostname

public-ipv4

public-keys/

reservation-id

security-groups

AWS CLI :

IAM User :

--> Management console access

--> programatic access : AccessKeyId and SecretAccessKey

https://aws.amazon.com/cli

aws --version

aws servicename commands

aws s3 ls

aws s3 ls s3://bucketname

copy:

aws s3 cp source destination

Task : Launch an ec2 instance using CLI. Terminate instance using CLI.

To launch the ec2 instance

**Step 1:**

C:\Users\Mani> **aws configure**

provide the Access key for the user who has all permissions.

**Step 2:** create a key pair

Aws ec2 describe-key-pairs

C:\Users\Mani>**aws** **ec2 create-key-pair --key-name MyKeyPair --query 'KeyMaterial' --output text > MyKeyPair.pem**

aws ec2 create-key-pair --key-name mykey1 --region ap-south-1 --output text

C:\Users\Mani>**aws** **ec2 describe-key-pairs --key-name MyKeyPair**

{

"KeyPairs": [

{

"KeyPairId": "key-0ce8422960be59e86",

"KeyFingerprint": "63:aa:fc:f4:5d:93:50:d1:ef:40:ab:6f:f5:06:d6:1f:c6:4a:79:07",

"KeyName": "MyKeyPair",

"Tags": []

}

]

}

Step 3: create a security group

Aws ec2 describe-security-groups

It will display ALL available security groups in ec2.

Before creating the Security group we need to know the VPC-id, so use the

**Aws ec2 describe-vpcs**

We will get the vpc id

C:\Users\Mani>aws ec2 describe-vpcs

{

"Vpcs": [

{

"CidrBlock": "172.31.0.0/16",

"DhcpOptionsId": "dopt-98fe04f3",

"State": "available",

"VpcId": "vpc-ed6f7085",

"OwnerId": "693652578065",

"InstanceTenancy": "default",

"CidrBlockAssociationSet": [

{

"AssociationId": "vpc-cidr-assoc-e1ead989",

"CidrBlock": "172.31.0.0/16",

"CidrBlockState": {

"State": "associated"

}

}

],

"IsDefault": true

}

]

}

Now create the Security group:

C:\Users\Mani>**aws ec2 create-security-group --group-name first-sg --description "first security group" --vpc-id vpc-ed6f7085**

**Step 4:**

Security group has created, no we need to add the rules, inbound rules and outbound rules.

Now we are going to add the rules for it.

Now make sure security group id is ready in handy,

[<http://checkip.amazonaws.com/> ] by using this we can check the ip address.

[ C:\Users\Mani>aws ec2 authorize-security-group-ingress --group-id sg-07384901b6b2f683a --protocol tcp --port 22 --cidr 103.252.26.238/32

C:\Users\Mani> ]

**Step 5:**

We need AMI id : [Amazon Machine Image ] ami-0732b62d310b80e97

Security group id: [sg-07384901b6b2f683a](https://ap-south-1.console.aws.amazon.com/ec2/v2/home?region=ap-south-1#SecurityGroup:groupId=sg-07384901b6b2f683a)

Subnet id : "SubnetId": "subnet-0c404464" to know this run the command aws ec2 describe-subnets

aws ec2 run-instances --image-id ami-xxxxxxxx --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group-ids sg-903004f8 --subnet-id subnet-6e7f829e

[C:\Users\Mani>aws ec2 run-instances --image-id ami-0732b62d310b80e97 --count 1 --instance-type t2.micro --key-name MyKeyPair --security-group-ids sg-07384901b6b2f683a --subnet-id subnet-0c404464 ]

Connect to instance

aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro" --query "Reservations[].Instances[].InstanceId"

**------------ launch the instnace ----------------**

aws ec2 run-instances --image-id ami-0732b62d310b80e97 --count 1 --instance-type t2.micro --key-name Linuxkeypair --security-group-ids sg-0d4df9011ed66374e --subnet-id subnet-0c404464

**---------- tag creation for instance -----------**

aws ec2 create-tags --resources i-0f381fdc1a30f8238 --tags Key=Name,Value=MyEC2FromCLI

**-----------listing all instance form an instnace type -------------------------**

aws ec2 describe-instances --filters "Name=instance-type,Values=t2.micro" --query Reservations[].Instances[].InstanceId

**----------List all instance launched with a specific AMI -----------------**

aws ec2 describe-instances --filters "Name=image-id,Values=,,ami-xxxxxxxxxxxx,ami-yyyyyyyyyyy,ami-zzzzzzzzzz>>" --query Reservations[].Instances[].InstanceId

**-------------list all instance with a specified tag ----------------**

aws ec2 describe-instances --filters "Name=tag:<<Tag Name>>,Values=<<instancename>>"

**---------- stop ec2 instance --------**

aws ec2 stop-instances --instance-ids i-0f381fdc1a30f8238

**-------------------- sstart ec2 instance -------**

aws ec2 start-instances --instance-ids i-0f381fdc1a30f8238

**------------ terminate instance ----------**

aws ec2 terminate-instances --instance-ids i-0f381fdc1a30f8238

**access key :** AKIA2DAHKL4IQMVFUJHV

**aceess key pwd :** tW/vVYfkZxSP6OO6zBGJrnhDU7JZQVJ+MGz1lLaX

**instance ami name :** ami-0732b62d310b80e97

-----------------------------------------------------------------------------------------------------