EC2 to S3 communication - IAM Role

Cloudformation

An IAM role is an IAM identity that you can create in your account that has specific permissions. An IAM role is similar to an IAM user, in that it is an AWS identity with permission policies that determine what the identity can and cannot do in AWS. However, instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it. Also, a role does not have standard long-term credentials such as a password or access keys associated with it. Instead, when you assume a role, it provides you with temporary security credentials for your role session.

Actually what happens is, when you create an IAM Role for EC2 using the IAM Console, it creates both an EC2 instance profile as well as an IAM role with same name.

So ideally, when you launch an instance with an IAM role, you get your instance profile list in the drop-down and you choose one of them for you.

When you are using the AWS CLI, SDKs, or CloudFormation, you will need to define both of them explicitly.

An IAM role with policies and permissions,
An EC2 instance profile containing a role

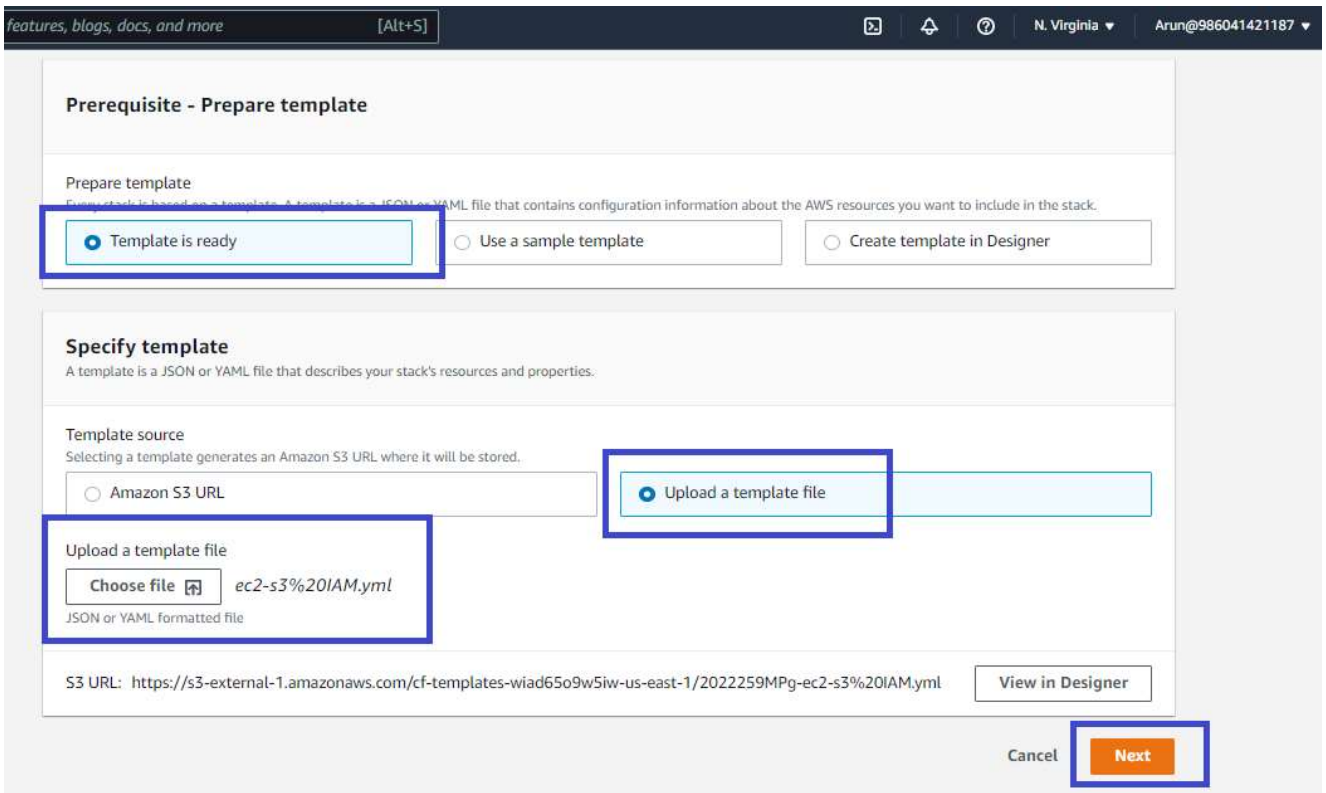
This time role name and instance profile name can be different so make sure that you use instance profile name while attaching to an EC2 instance.

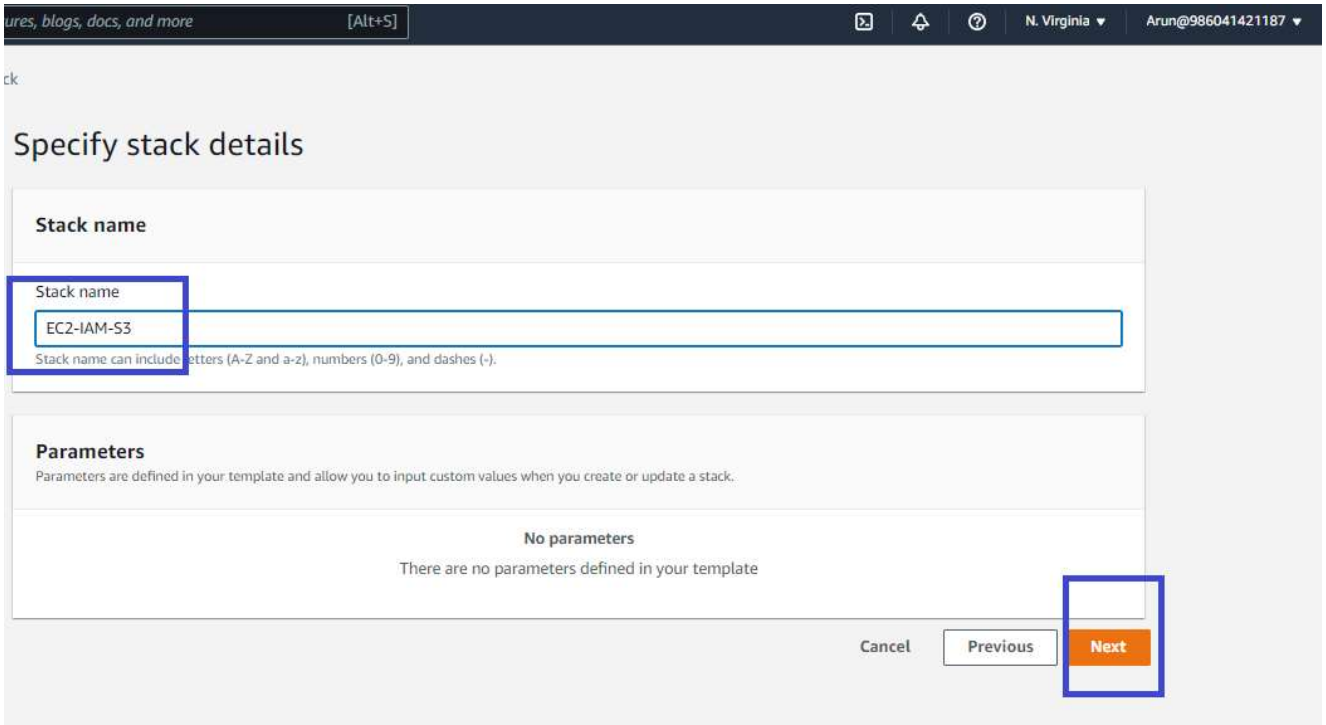
YAML FILE

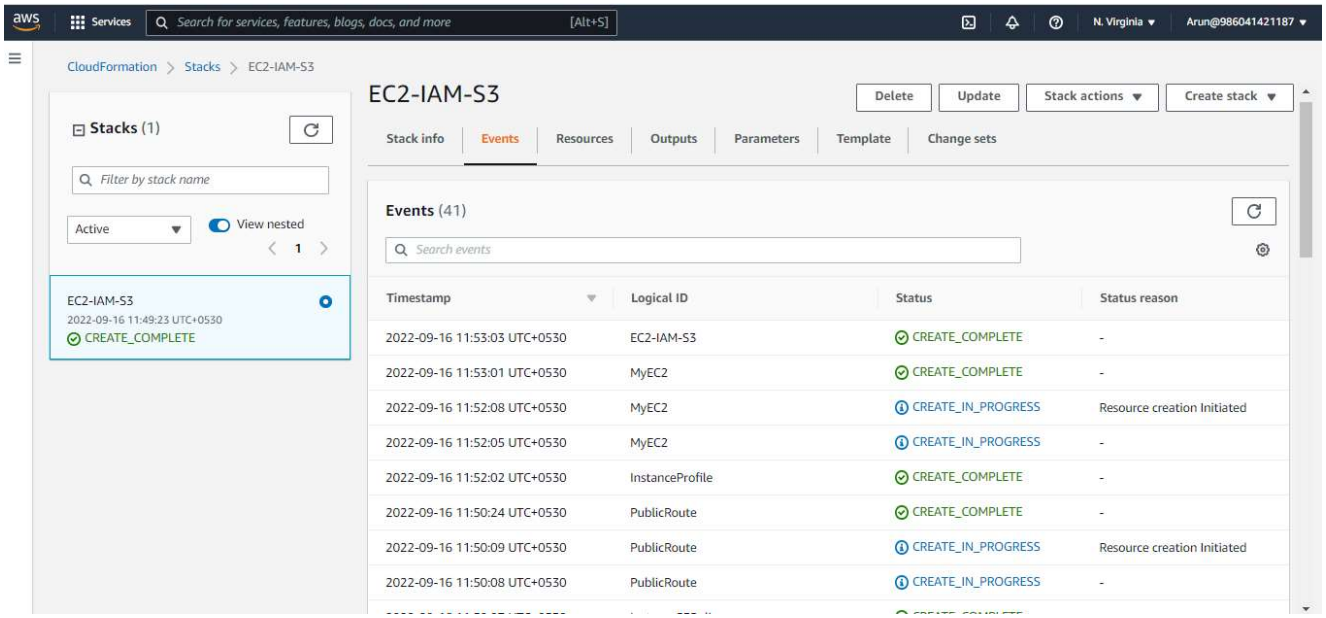
```
Resources:
  MyVPC:
    Type: AWS::EC2::VPC
    Properties:
      CidrBlock: 192.178.0.0/16

  PublicSubnet:
    Type: AWS::EC2::Subnet
    Properties:
      VpcId: !Ref MyVPC
      CidrBlock: 192.178.1.0/24
      MapPublicIpOnLaunch: "true"
      AvailabilityZone: "us-east-1a"

  IntGateway:
    Type: AWS::EC2::InternetGateway

  Attachgateway:
    Type: AWS::EC2::VPCGatewayAttachment
    Properties:
      VpcId: !Ref MyVPC
      InternetGatewayId: !Ref IntGateway

  PublicRouteTable:
      Type: AWS::EC2::RouteTable
      Properties:
        VpcId: !Ref MyVPC

  PublicRoute:
      Type: AWS::EC2::Route
      DependsOn: Attachgateway
      Properties:
        RouteTableId:
          Ref: PublicRouteTable
        DestinationCidrBlock: 0.0.0.0/0
        GatewayId: !Ref IntGateway

  PublicSubnetRouteTableAssociation:
      Type: AWS::EC2::SubnetRouteTableAssociation
      Properties:
        SubnetId: !Ref PublicSubnet
        RouteTableId: !Ref PublicRouteTable

  Securitygroup:
```

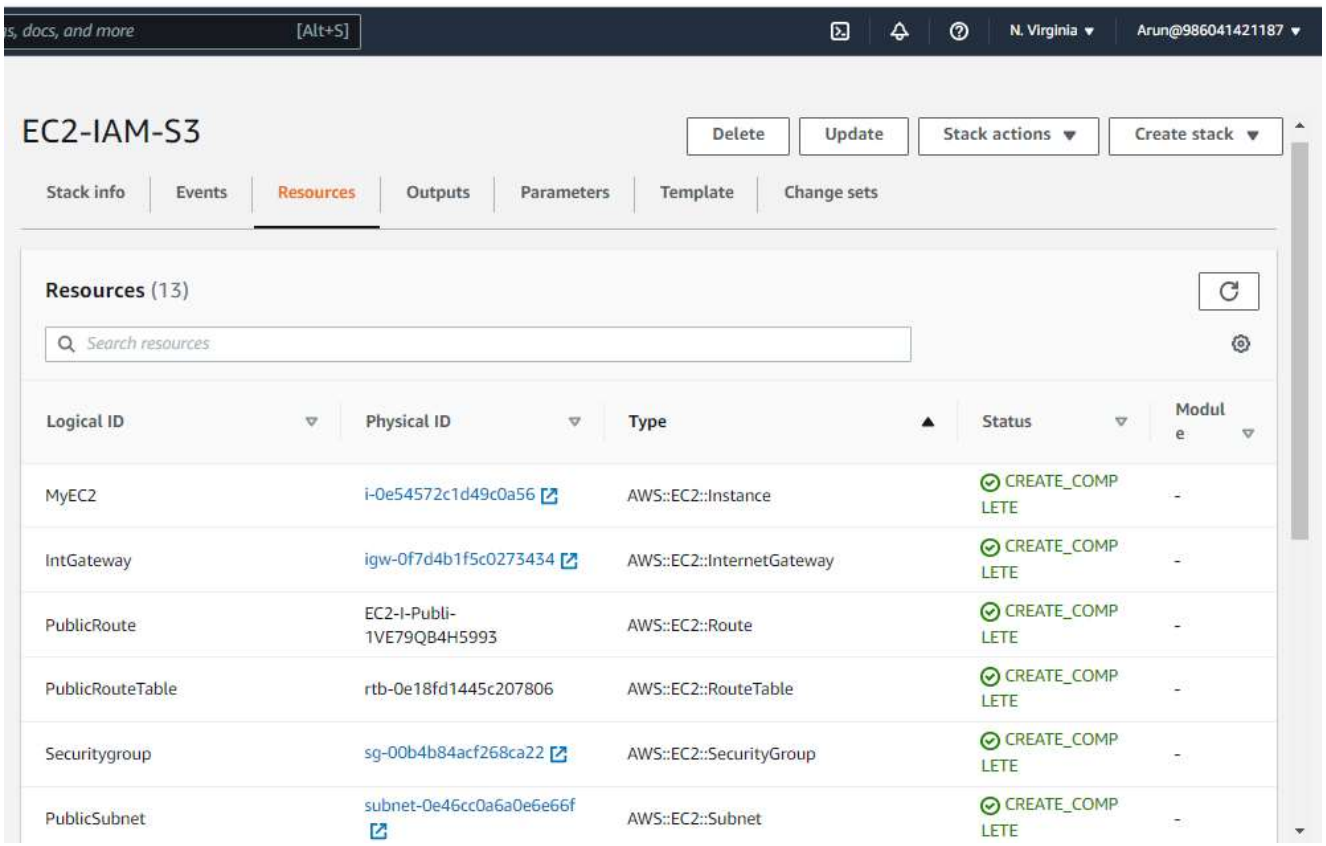
```
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: Allow SSH and Mysql
    SecurityGroupIngress:
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        CidrIp: 0.0.0.0/0

      - IpProtocol: tcp
        FromPort: 3306
        ToPort: 3306
        CidrIp: 0.0.0.0/0
    VpcId: !Ref MyVPC

MyEC2:
  Type: AWS::EC2::Instance
  Properties:
    ImageId: "ami-052efd3df9dad4825"
    InstanceType: "t2.micro"
    KeyName: "08-09-2022"
    SecurityGroupIds:
      - !Ref Securitygroup
    SubnetId: !Ref PublicSubnet
    IamInstanceProfile: !Ref InstanceProfile
    UserData:
      Fn::Base64: !Sub |
        #! /bin/bash
        sudo apt update
        sleep 20
        apt install awscli

InstanceProfile:
  Type: AWS::IAM::InstanceProfile
  Properties:
    InstanceProfileName: ec2-instance-profile
    Path: /
    Roles:
     - !Ref Ec2InstanceRole

Ec2InstanceRole:
  Type: AWS::IAM::Role
  Properties:
    RoleName: ec2-instance-role
    AssumeRolePolicyDocument:
      Version: 2012-10-17
```

```
      Statement:
       -
         Effect: Allow
         Principal:
           Service:
             - ec2.amazonaws.com
         Action:
           - sts:AssumeRole
     Path: /

  InstanceS3Policy:
    Type: AWS::IAM::Policy
    Properties:
      PolicyName: DemoS3Policy
      PolicyDocument:
        Version: 2012-10-17
        Statement:
         -
           Effect: Allow
           Action:
             - s3:*
           Resource: "*"
      Roles:
        -
          !Ref Ec2InstanceRole

  S3Bucket:
    Type: AWS::S3::Bucket
    Description: Creating Amazon S3 bucket from CloudFormation
    Properties:
      BucketName: i-named-this-bucket-16-09-22

Outputs:
  S3Bucket:
    Description: Bucket Created using this template.
    Value: !Ref S3Bucket
  VPCId:
    Description: "VPCId of VPC"
    Value: !Ref "MyVPC"

  PublicIP:
    Description: Public IP of EC2 Instance
    Value: !GetAtt MyEC2.PublicIp
```

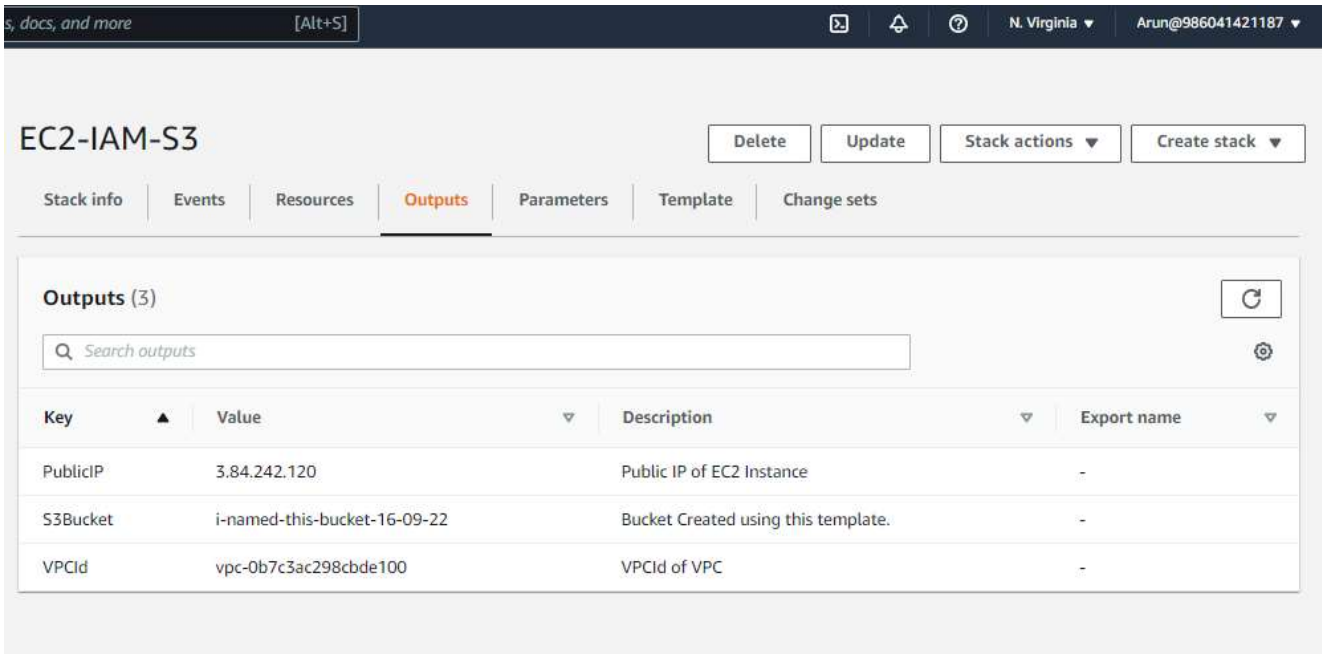

features, blogs, docs, and more [Alt+S]
N. Virginia
Arun@986041421187
Prerequisite - Prepare template
Prepare template
Every stack is based on a template. A template is a JSON or YAML file that contains configuration information about the AWS resources you want to include in the stack.
Template is ready
Use a sample template
Create template in Designer
Specify template
A template is a JSON or YAML file that describes your stack's resources and properties.
Template source
Selecting a template generates an Amazon S3 URL where it will be stored.
Amazon S3 URL
Upload a template file
Upload a template file
Choose file
ec2-s3%20IAM.yml
JSON or YAML formatted file
S3 URL: https://s3-external-1.amazonaws.com/cf-templates-wiad65o9w5iw-us-east-1/2022259MPg-ec2-s3%20IAM.yml
View in Designer
Cancel
Next


ures, blogs, docs, and more [Alt+S]
N. Virginia
Arun@986041421187
ck
Specify stack details
Stack name
Stack name
EC2-IAM-S3
Stack name can include letters (A-Z and a-z), numbers (0-9), and dashes (-).
Parameters
Parameters are defined in your template and allow you to input custom values when you create or update a stack.
No parameters
There are no parameters defined in your template
Cancel
Previous
Next


Services
Search for services, features, blogs, docs, and more [Alt+S]
N. Virginia
Arun@986041421187
CloudFormation > Stacks > EC2-IAM-S3
Stacks (1)
Filter by stack name
Active
View nested
EC2-IAM-S3
2022-09-16 11:49:23 UTC+0530
CREATE_COMPLETE
EC2-IAM-S3
Delete
Update
Stack actions
Create stack
Stack info
Events
Resources
Outputs
Parameters
Template
Change sets
Events (41)
Search events
Timestamp
Logical ID
Status
Status reason
2022-09-16 11:53:03 UTC+0530 EC2-IAM-S3 CREATE_COMPLETE -
2022-09-16 11:53:01 UTC+0530 MyEC2 CREATE_COMPLETE -
2022-09-16 11:52:08 UTC+0530 MyEC2 CREATE_IN_PROGRESS Resource creation Initiated
2022-09-16 11:52:05 UTC+0530 MyEC2 CREATE_IN_PROGRESS -
2022-09-16 11:52:02 UTC+0530 InstanceProfile CREATE_COMPLETE -
2022-09-16 11:50:24 UTC+0530 PublicRoute CREATE_COMPLETE -
2022-09-16 11:50:09 UTC+0530 PublicRoute CREATE_IN_PROGRESS Resource creation Initiated
2022-09-16 11:50:08 UTC+0530 PublicRoute CREATE_IN_PROGRESS -


[Alt+S]
N. Virginia
Arun@986041421187
EC2-IAM-S3
Delete
Update
Stack actions
Create stack
Stack info
Events
Resources
Outputs
Parameters
Template
Change sets
Resources (13)
Search resources
Logical ID
Physical ID
Type
Status
Module
MyEC2 i-0e54572c1d49c0a56 AWS::EC2::Instance CREATE_COMPLETE -
IntGateway igw-0f7d4b1f5c0273434 AWS::EC2::InternetGateway CREATE_COMPLETE -
PublicRoute EC2-I-Publi-1VE79QB4H5993 AWS::EC2::Route CREATE_COMPLETE -
PublicRouteTable rtb-0e18fd1445c207806 AWS::EC2::RouteTable CREATE_COMPLETE -
Securitygroup sg-00b4b84acf268ca22 AWS::EC2::SecurityGroup CREATE_COMPLETE -
PublicSubnet subnet-0e46cc0a6a0e6e66f AWS::EC2::Subnet CREATE_COMPLETE -

PublicSubnetRouteTableAssociation	rtbassoc-0cab16de6ba225df9	AWS::EC2::SubnetRouteTableAssociation	CREATE_COMPLETE	-
MyVPC	vpc-0b7c3ac298cbde100	AWS::EC2::VPC	CREATE_COMPLETE	-
Attachgateway	EC2-I-Attac-ZHGBJTVX1EZ5	AWS::EC2::VPCGatewayAttachment	CREATE_COMPLETE	-
InstanceProfile	ec2-instance-profile	AWS::IAM::InstanceProfile	CREATE_COMPLETE	-
InstanceS3Policy	EC2-I-Inst-1NS50849BXN5Z	AWS::IAM::Policy	CREATE_COMPLETE	-
Ec2InstanceRole	ec2-instance-role	AWS::IAM::Role	CREATE_COMPLETE	-
S3Bucket	i-named-this-bucket-16-09-22	AWS::S3::Bucket	CREATE_COMPLETE	-



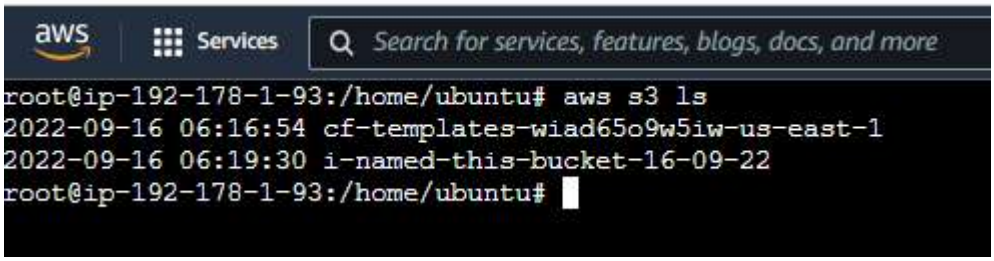
Now Connect to EC2

Check AWS CLI Version

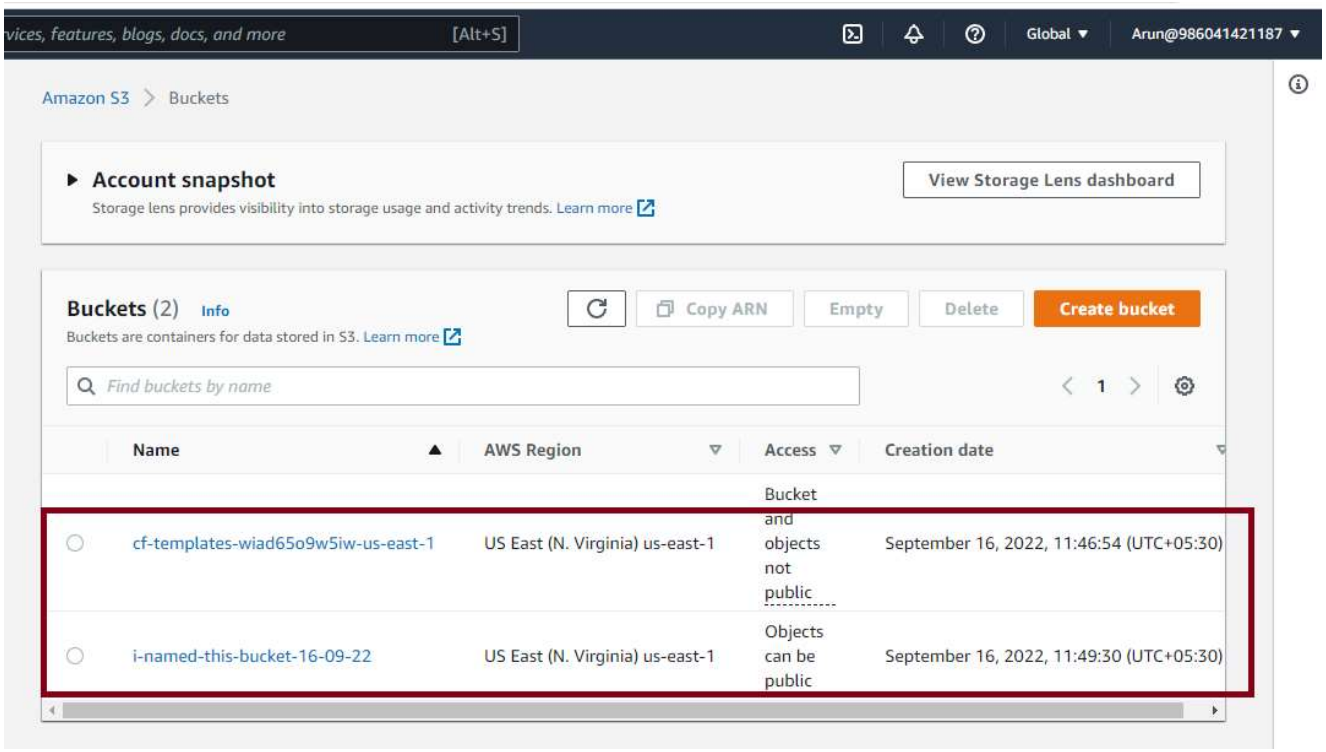
```
root@ip-192-178-1-93:/home/ubuntu# aws --version
aws-cli/1.22.34 Python/3.10.4 Linux/5.15.0-1011-aws botocore/1.23.34
root@ip-192-178-1-93:/home/ubuntu#
```

Then apply command aws s3 ls

we get the file in s3 bucket



we can check this manually with management console.



Hence Successfully Performed