Assignment 3 – S3

Always follow the least privilege principle for the IAM policies and security groups.

Demo Elastic IP

1. S3 and EC2 – Download a file (index.html) from S3 in EC2 instances using identity-based and resource-based policies. That is why you need to create 2 buckets and 2 EC2 instances.
    a. Create two S3 buckets and put index.html.
    b. Associate the LabRole while creating the first instance.
    c. Don't associate the LabRole while creating the second instance. SSH into that using Key Pair. Windows users need PuTTy software. MacOS users just use your terminal.
    d. Create an inline IAM policy in the LabRole that allows the first instance to get objects from the bucket. [**Include it in the PDF**]
    e. Create a resource-based policy on the second S3 that allows the second EC2 instance to get objects. [**Include it in the PDF**]
    f. Navigate to /var/www/html folder. Download the index.html in S3 in the EC2. Copy it from S3 to EC2.

aws s3 cp s3://<bucket_name>/<file_name_in_s3> <new_file_name_in_EC2>

2. S3 event notification – Send an email to yourself when the object is created in the bucket.
    a. You need to create an SNS topic.
    b. Write a resource-based policy that allows S3 to send messages to the topic. Modify the default SNS policy while creating the SNS. [**Include it in the PDF**]
    c. Subscribe to it with your email.
3. S3 signed URL – Write a lambda that returns a Signed URL of the object in S3. Make sure the LabRole has an inline policy that allows getting objects from the bucket.

```
const AWS = require("aws-sdk");
const s3 = new AWS.S3({apiVersion: '2006-03-01'});

exports.handler = async (event) => {
    const params = { Bucket: 'myfirstbucketcreatedwithcli2022cs516', Key:
'Capture.PNG' };
    return s3.getSignedUrl('getObject', params);
};
```

Refer: https://docs.aws.amazon.com/AWSJavaScriptSDK/latest/AWS/S3.html#getSignedUrl-property

Extra:

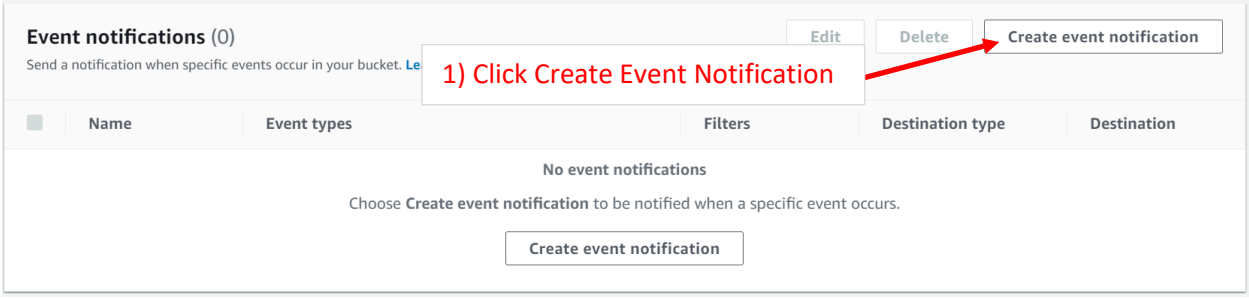
- Read a file in S3 in EC2 using S3 Gateway Endpoint. After a successful connection, write S3 resource-based policy that allows reading access only from the VPC endpoint in the bucket policy. Refer: Amazon S3 and VPC Endpoints

Setting Event Notification

Create SNS Topic & Add Permission for S3 to Publish Message. Subscribe to the SNS Topic

```
{
      "Effect": "Allow",
      "Principal": {
        "Service": "s3.amazonaws.com"
      },
      "Action": "SNS:Publish",
      "Resource": "<Your SNS ARN>",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "<Your AWS Account Number>"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:s3:*:*:<Your Bucket Name>"
        }
      }
    }
```

Create S3 Event Notification for Your Bucket Under Properties Tab Event Notification



...

Create event notification Info

The notification configuration identifies the events you want Amazon S3 to publish and the destinations where you want Amazon S3 to send the notifications. Learn more

General configuration

2) Name Event

Event name

S3_Put_Notification

Event name can contain up to 255 characters.

Prefix - *optional*

Limit the notifications to objects with key starting with specified characters.

images/

Suffix - *optional*

Limit the notifications to objects with key ending with specified characters.

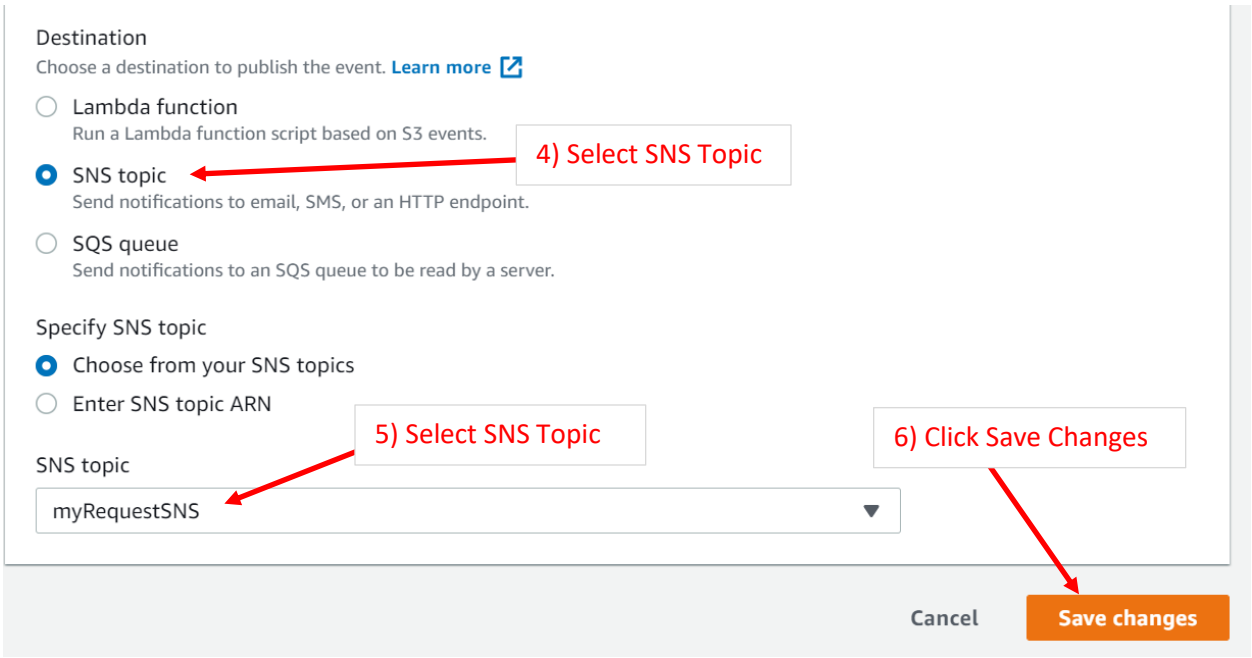
.jpg

Event types

Specify at least one type of event for which you want to receive notifications. Learn more

- [ ] All object create events
  s3:ObjectCreated:*
  - [x] Put
    s3:ObjectCreated:Put
  - [ ] Post
    s3:ObjectCreated:Post
  - [ ] Copy
    s3:ObjectCreated:Copy
  - [ ] Multipart upload completed
    s3:ObjectCreated:CompleteMultipartUpload
- [ ] All object delete events
  s3:ObjectRemoved:*
  - [ ] Permanently deleted
    s3:ObjectRemoved:Delete

3) Select Your Criteria

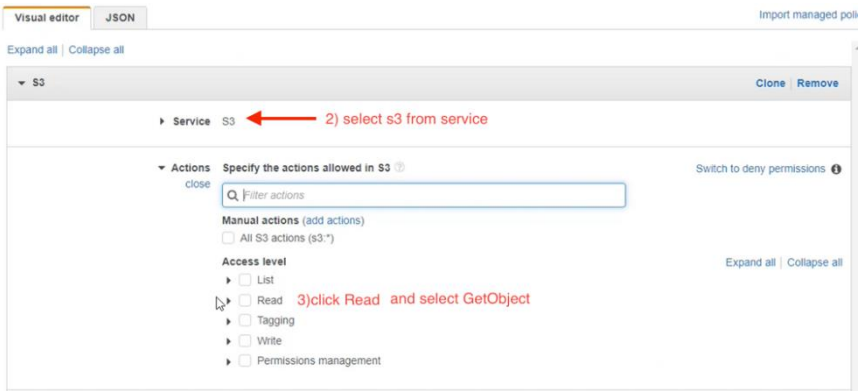


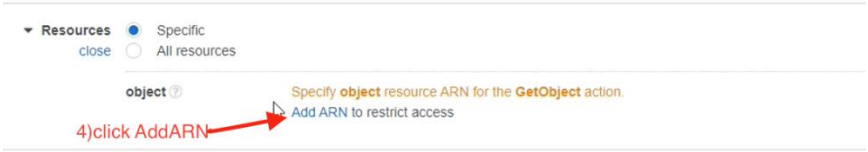
Test by Uploading a File to Your S3 Bucket

Downloading the index HTML from S3 in EC2

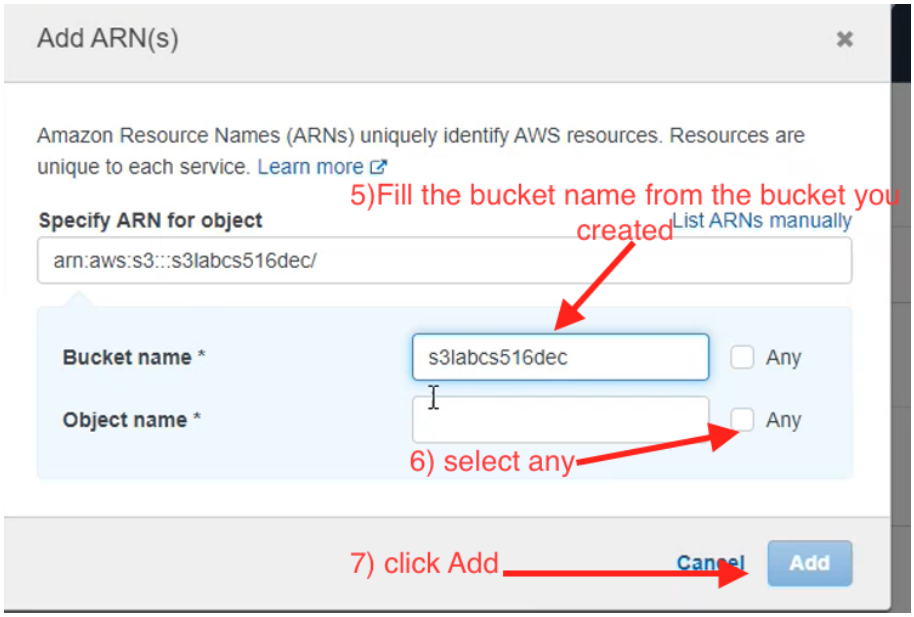
1) Creating a bucket for assets of the web app hosted on EC2.

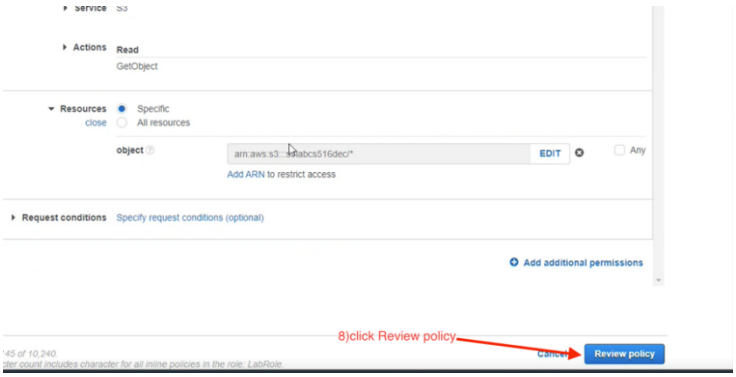
First go to IAM-->Role-->LabRole



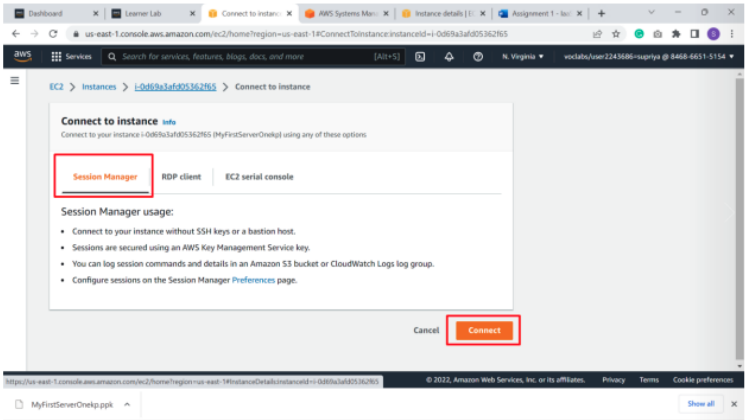


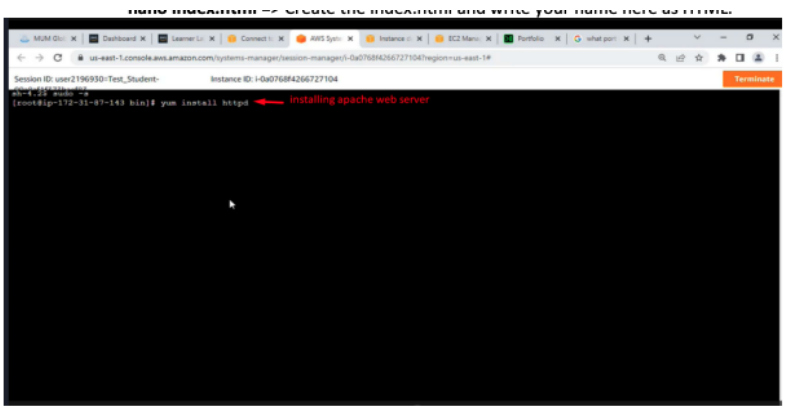
Create a bucket and grab(copy) the backet name.

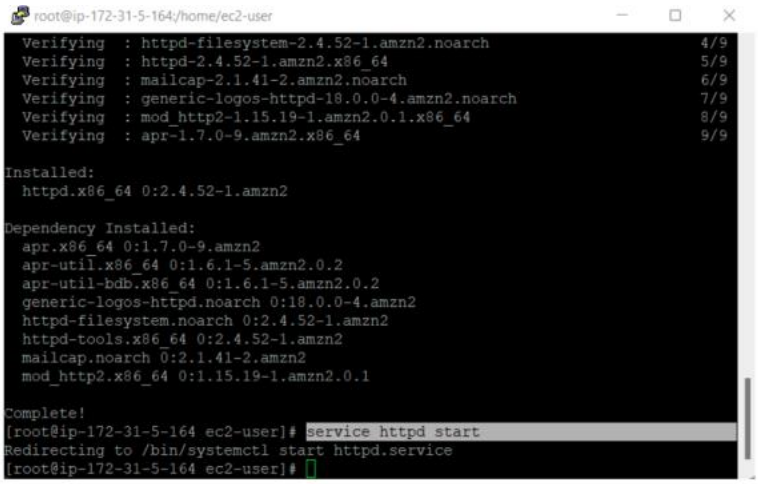


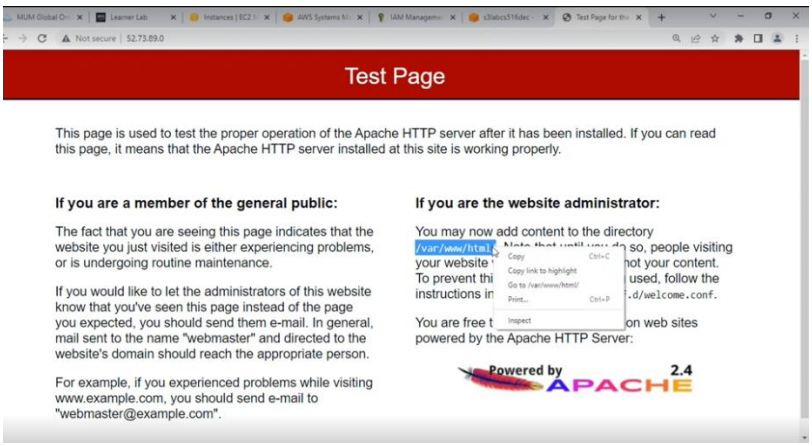


Create an index.html file and upload it to the bucket created earlier.
Go to EC2 service, connect the instance which is associated with LabRole or create one.


EC2 > Instances > i-0d69a3afd05362f65 > Connect to instance
Connect to instance Info
Connect to your instance i-0d69a3afd05362f65 (MyFirstServerOnekp) using any of these options
Session Manager
RDP client
EC2 serial console
Session Manager usage:
Connect to your instance without SSH keys or a bastion host.
Sessions are secured using an AWS Key Management Service key.
You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
Configure sessions on the Session Manager Preferences page.
Cancel
Connect


Session ID: user2196930=Test_Student-
Instance ID: i-0a0768f4266727104
Terminate
[root@ip-172-31-87-143 bin]# yum install httpd
Installing apache web server


root@ip-172-31-5-164:/home/ec2-user
Verifying : httpd-filesystem-2.4.52-1.amzn2.noarch 4/9
Verifying : httpd-2.4.52-1.amzn2.x86_64 5/9
Verifying : mailcap-2.1.41-2.amzn2.noarch 6/9
Verifying : generic-logos-httpd-18.0.0-4.amzn2.noarch 7/9
Verifying : mod_http2-1.15.19-1.amzn2.0.1.x86_64 8/9
Verifying : apr-1.7.0-9.amzn2.x86_64 9/9
Installed:
httpd.x86_64 0:2.4.52-1.amzn2
Dependency Installed:
apr.x86_64 0:1.7.0-9.amzn2
apr-util.x86_64 0:1.6.1-5.amzn2.0.2
apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2
generic-logos-httpd.noarch 0:18.0.0-4.amzn2
httpd-filesystem.noarch 0:2.4.52-1.amzn2
httpd-tools.x86_64 0:2.4.52-1.amzn2
mailcap.noarch 0:2.1.41-2.amzn2
mod_http2.x86_64 0:1.15.19-1.amzn2.0.1
Complete!
[root@ip-172-31-5-164 ec2-user]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-5-164 ec2-user]#



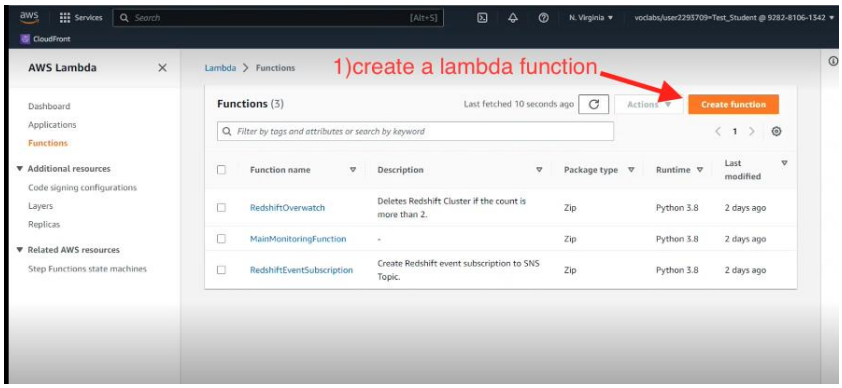
This line of code will download the image. aws s3 cp s3://<bucket>/ index.html indx.html

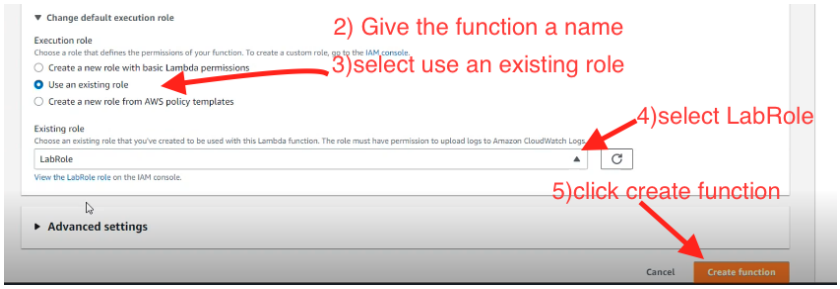
```
[root@ip-172-31-82-225 bin]# service httpd start
Redirecting to /bin/systemctl start httpd.service
[root@ip-172-31-82-225 bin]# cd /var/www/html/
[root@ip-172-31-82-225 html]# ls
[root@ip-172-31-82-225 html]# aws s3 cp s3://s3labcs516dec/index.html index.html
download: s3://s3labcs516dec/index.html to ./index.html
[root@ip-172-31-82-225 html]# ls
index.html
[root@ip-172-31-82-225 html]#
```

Lambda app for getting the presigned URL for an object in S3

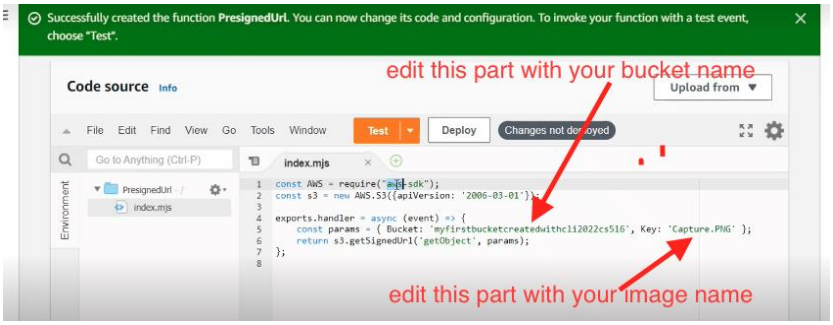
2). Writing a lambda that returns a Signed URL of the object and making sure that the LabRole has an inline policy that allows getting objects from the bucket.

Creating lambda function.

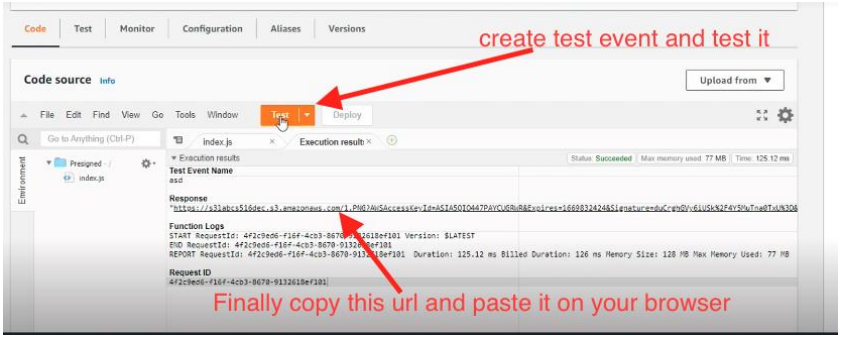


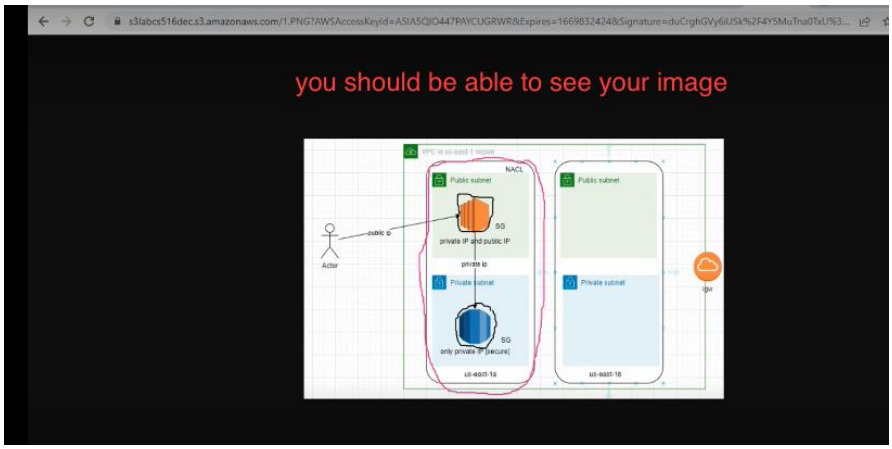


Writing the function.



Testing the function.




s3labcs516dec.s3.amazonaws.com/1.PNG?AWSAccessKeyId=ASIA5QIO447PAYCUGRWR&Expires=1669832424&Signature=duCrghGVy6iUSk%2F4Y5MuTna0TxU%3...
you should be able to see your image
NACL
Public subnet
Public subnet
SG
private IP and public IP
public ip
Actor
private ip
Private subnet
Private subnet
igw
SG
only private IP (secure)
us-east-1a
us-east-1b