7(a)

Aim:: Creating a VPC Peering Connection

Description:: You might want to Connect your virtual private clouds (VPCs) when you must transfer data between them. This lab shows you create a private VPC peering connected between two VPCs.

After Completing this lab, you should be able to:

- create a VPC peering Connection
- Configure route tables to use the VPC peering connection.

Task 1:: creating a VPC Peering Connection

Your task is to Create a VPC peering Connection between two VPCs.

* In the AWS management Console, on the services menu, Choose VPC.
* In the left navigation pane, choose Peering Connections.
* Choose Create Peering Connection and Configure:
    - peering Connection name tag : Lab-peer
    - VPC (Requestor) : Lab VPC
    - VPC (Accepter) : Shared VPC
    - choose Create Peering Connection then choose ok.
* select Lab-Peer
* choose Actions then select Accept Request, and choose Yes

  Accept to accept the request
* In the pop-up box, choose Close

Task 2 :- Configuring route tables

* In the left navigation pane, Choose Route Tables.
* Select Lab Public Route table (for Lab VPC)
* In the Routes tab, choose Edit routes then configure these settings:
  - Choose Add route
  - Destination : 10.5.0.0/16
  - Target : Select Peering Connection,
  - Choose Save routes the choose Close
* Select shared-VPC Route Table
* In the Routes tab, Choose Edit routes then Configures these settings :
  - Choose Add route
  - Destination : 10.0.0.0/16
  - Target : Select Peering Connection
  - Choose Save routes then Choose Close.

Task 3 : Testing the VPC Peering Connection

Now that you Configured VPC peering, you will test the VPC peering Connection.

* on the services menu, Choose EC2
* on the left navigation pane, Choose Instances
* copy the IPV4 Public IP address that is shown in Description
* Open a new web browser tab with that IP address.
* Choose settings
  - Endpoint
  - Database : inventory
  - Username : admin
  - Password : lab-password
  - Choose Save.

Submit your work

* At the top of these instruction choose Submit
* Choose End lab at the top of this page.

9 (a)

Aim:- Creating a Highly Available Environment

Description:- In this lab environment, access to Aws services and service actions might be restricted to the ones that are needed to complete the lab instructions. You might encounter errors if you attempt to access other services or perform actions beyond the ones that are described in this lab.

Task 1:

* On the Aws management Console, on services menu choose VPC.
* under filter by VPC, select a VPC box and select lab VPC.
* Choose Your VPCs
* Choose subnets
* select Public Subnet 1.
* Choose the Route table tab
* Choose the Network ACL tab
* In the left Navigation pane, choose Internet gateways
* Choose Security groups
* select Inventory-DB.
* Choose the outbound rules tab.

Task 2:

* On the services menu, choose EC2
* Choose Load Balancers
* Choose create load balancer
* under Application Load Balancer, Choose Create
* for load balancer name, enter: Inventory-LB

* scroll down to the Network mapping section, then for VPC, select lab VPC
* Under mappings, Choose the first Availability zone, then choose the public subnet that displays
* Choose the second Availability zone, then choose Public subnet that displays
* In the security groups.
  - security group : Inventory-LB
  - Description : Enable web access to load balancer
  - VPC : Select Lab VPC.
* Under Inbound rules, Choose Add rule and configure
  - Type: HTTP
  - Source : Anywhere - IPv4
* still under Inbound rules, choose Add rule
  - Type: HTTPS
  - Source : Anywhere - IPv4
* choose Create security group.
* configure the target group
  - Choose a target type : Instances
  - Target group name : Inventory-App
  - VPC : Lab VPC
  - scroll down and expand Advanced health check settings
  - Healthy threshold : 2
  - Interval : 10
  - Choose Next.
  - Review the settings & choose Create target group.

Task 3 : Creating an Auto Scaling group

* In the AWS Console, Choose EC2
* Choose Instances
* select web server 1

* Choose Image & templates > Create image.
  - Image Name: web server AMI
  - Image description: Lab AMI for Webserver
* Choose Create Image.

Create a Launch Template and an Auto scaling Group

* Choose Launch Templates in left navigation pane.
* Choose Create launch template
* Configure the launch template settings & create it:
  - Launch template name: Inventory-LT
  - AMI: Webserver AMI
  - Instance type: Choose t2.micro
  - Key pair name: Choose vockey
  - Firewall: select existing security group
  - Security groups: Choose Inventory-App
* Choose Inventory-LT in Success dialog.
* From Actions menu, Choose Create Auto scaling group
*
  - Autoscaling group name: Inventory-ASG
  - Launch template: Confirm
  - Choose Next
*
  - VPC: Choose Lab VPC
  - Choose Next
* Cofigure the detains in steps
  - Choose Next
*
  - Key: Name
  - value: Inventory-App

Task 4: Updating security groups

* Choose Security Groups
* Select Inventory-App.
* Choose the Inbound rules tab.
* Choose Edit Inbound rules
* Choose Add rule
  - Type: HTTP
  - Description: Traffic from loadbalancer
  - Choose save rules.
* choose Edit Inbound rules & Configure these settings:
  - Delete the existing rule
  - Choose Add rule
  - for Type, choose MYSQL/Aurora
  - Choose the search box to the right of Custom
  - Enter sg
  - Select Inventory-App
  - Choose save rules

Task 5: Testing the application

* choose Target Groups
* select Inventory-App
* choose the Targets tab.
* Choose load balancers and then choose Inventory-LB.
* Copy the DNS name to your clipboard.
* Reload the page in your web browser.

Task 6: Testing high availability

* Return to EC2 Console tab in your web browser.

* Choose Instances
* Select one of the Inventory-App
* Choose Instance state > Terminate instance.
* Choose Terminate.

Submitting your work

* Choose Submit to record your progress and when prompted, choose Yes.
* Choose End Lab

10(a)

Aim:- Automating Infrastructure Deployment with Aws CloudFormation

Description:- In this lab, you will learn how to deploy multiple layers of infrastructure with Aws cloudformation, update a cloud formation stack, and delete a stack

Task 1:- Deploying a networking layer

* Right-click the following link and download the template to your Computer: lab-network.yaml
* Choose cloudformation from services menu
* Choose create stack & configure these settings

Step 1: Specify template

- Template source: upload
- Upload a template file: click choose file
- Choose Next

Step 2: Create stack

- Stack name: lab-network
- Choose Next

Step 3: Configure stack options

- Key: application
- value: inventory
- Choose Next.

Step 4: Review lab-network

- Choose create stack.

* Choose the stack info tab
* wait for the status to change to CREATE_COMPLETE
* Choose the Resources tab.

* Choose the Events tab
* Choose the Outputs tab
* Choose the template tab.

Task 2: Deploying an application layer

* lab-application.yaml Right-click & download the template
* Choose stacks
* Select Create stack > with new resources

step 1: specify template.

- Template source: Upload
- Upload a template file: click Choose file

step 2: create stack

- Stack name: lab-application
- Network stack Name: lab-network
- Choose Next.

step 3:-

- Key: application
- value: inventory
- Choose Next

step 4:-

- Choose Create stack.

* In the stack info tab, wait for the status to change to CREATE_COMPLETE.
* choose the Outputs tab.
* copy the URL that is displayed.

Task 3: Updating a stack

* In the AWS Management Console, from Services menu, Choose EC2
* Choose Security Groups.
* Select the check box for lab-application-WebServer Security Group
* Choose the Inbound rules tab.

* from the services menu, Choose CloudFormation.
* Right click on the link & download the updated template lab-application2.Yaml.
* Select lab-application
* Choose [Update]
  - select Replace current template
  - Template source: [Upload a template file]
  - Upload a template file: click [choose file]
* choose Next
* Choose Update stack.

Task4:- Exploring templates with AWS CloudFormating Designer

* Choose cloudformation
* Choose Designer.
* Choose file menu, select open> Local file & select the lab-application2.Yaml

Task 5:- Deleting the stack

* In the list of stacks, choose the lab-application link.
* choose [Delete]
* Choose [Delete stack]
* From the services menu, Choose EC2
* Choose snapshots.

Submitting your work

* Choose [Submit] to record your progress and when prompted, choose Yes
* Choose [End Lab]

11(a)

Aim :- Streaming Dynamic Content using Amazon CloudFront.

Description :- In this lab, you will use Amazon CloudFront to deliver a dynamic, multiple bit-rate stream to a connected device using Apple's HTTP Live Streaming protocol. The stream can be played on any browser that supports the HLS protocol. In this lab, you will also use Amazon Elastic Transcoder to convert a source video into multiple bit-rates that will be delivered using CloudFront.

Task 1 : Lab Preparation

* In the AWS Management Console, on the services menu, Choose S3.
* Open the Input folder. It contains a video file named Amazon S3 Sample.mp4.

Task 2 : Create an Amazon CloudFront Distribution

* on the services menu, Choose CloudFront.
* Choose Create a CloudFront distribution.
* under Origin settings
    - select the Origin domain
    - Leave Origin access as Public
    - Under WAF (web application firewall) select Do not enable security protections
* Choose Create Distribution.

Task 3 : Create an Amazon Elastic Transcoder Pipeline

* On the services menu, Choose Elastic Transcoder
* on the pipelines page, Choose Create a new pipeline
* for the Pipeline Name, enter [Input Pipeline]

* for Input Bucket, select the awstrainingreinvent s3 bucket
* for IAM Role, under other roles, select AmazonElasticTranscoderRole
* ○ under Bucket, select the awstraingreinvent s3 bucket.
  ○ under Storage Class, select standard.
* Choose Create Pipeline.

Create a Job

* Choose Create a New Job on the Pipelines page.
* for pipeline, select Input Pipeline.
* for Output key Prefix, enter output/.
* for input key, select the input file labeled input/Amazons3sample.mp4.

Configure a playlist

* under playlists, Choose Add playlist, then Configure
  ○ master playlist name primary
  ○ playlist format: HLSV3
* Choose Create New Job.

Task 4:- Test playback of the Dynamic Stream

* on the services menu, Choose CloudFront
* select the Amazon CloudFront & verify the status changed to Enabled or not.
* Proceed to next step after status is Enabled.
* Copy the Distribution domain name & paste it into a text editor

Obtain the Playlist File Path

* on the services menu, Choose s3
* select the awstrainingreinvent s3 bucket.
* open the output folder & select the primary.m3u8 playlist file.

* Type the URL into the default browser of an ios or Android device.
* The stream should start playing on your device and dyramically request the relevant segments based on your bandwidth and CPU conditions.

Submitting your work

Choose [submit] and then click on [End Lab].

14

Aim:- Hybrid storage and Data Migration with Aws storage. Gateway file Gateway

Description:- In this lab, you will use the Aws storage Gateway file Gateway service to attach a Network file system (NFS) mount to an on-premises data store. You will then replicate that data to an s3 bucket in Aws. Additionally, you will configure advanced Amazon s3 features, like Amazon s3 lifecycle policies and Cross-Region replication.

Task1: Reviewing the lab architecture

This Lab environment uses a total of three Aws Regions. A Linux EC2 instance that emulates an on-premises server is deployed to the us-east-1 Region. The storage Gateway virtual appliance is deployed to the same Region as the Linux server.

The primary s3 bucket is created in the us-east-2 Region. Data from the Linux host is copied to the primary s3 bucket. This bucket can be called the source.

Task2:- Creating the primary & Secondary s3 buckets

* Choose s3 to open the s3 Console
* Choose Create bucket
  - Bucket Name: create a name that you can remember easily & that must be unique.
  - Region: us East (Ohio) us-east-2
  - Bucket Versioning: Enable

* Choose Create bucket
*
  - Bucket name: Any name that must be unique
  - Region: US West
  - Versioning: Enable

Task 3 Enabling Cross-Region replication

* Select the Management tab & select [Create replication rule]
* Configure the Replication rule:
  - Replication rule name: crr-full-bucket
  - Status Enabled
  - Source bucket:
    for choose a rule scope, select Apply to all objects in the bucket.
  - Destination
    - Choose a bucket in this account
    - choose [Browse S3]
    - select choose path
    - IAM role: S3-CRR-Role.
* Choose [Add files]
* wait for the file to upload, then Choose close.

Task 4:- Configuring the File Gateway and Creating a NFS file share

* In the Search box to the right of Services, search for and Choose Storage Gateway
* At the top-right of the Console, verify that curren Region is N. Virginia
* Choose create gateway

- Gateway name: file Gateway
- Gateway time zone: Choose GMT - 5:00 Eastern Time
- Gateway type: Amazon s3 file Gateway

* Configure the network & security group settings for the gateway
  - Next to Network settings, choose Edit, then Configure:

    VPC: on-prem-VPC

    subnet: on-prem-subnet

    Auto-assign public IP: Enable

* Finish creating the gateway
  - Choose view all instances
* Select your file Gateway instance, then in the Details tab below, locate the public IPv4 address & copy it.
* In the step3: Review and activate settings screen choose Activate gateway
* Configure the step 4: Configure gateway settings:
  - cloudwatch log group: Deactivate logging
  - cloudwatch alarms: No Alarm
  - wait for the local disks status to show that it finished Processing
  - choose Configure.
* Start Creating a file share.
  - Object Metadata
    - Guess MIME type
    - Gateway files accessible to s3 bucket owner
    - Enable Requestor Pays.
  - Choose Next.

Tasks: Mounting the file share to the linux instance and migrating the data

* Connect to the on-prem Linux Server instance

Microsoft Windows users

* Choose the [Details] & then select [show]

A credentials window opens.

* Note the On Prem Linux Instance address, if it is displayed.
* Exit the details panel by Choosing the x.
* Open putty.exe.
* To trust and Connect to the host, Choose Accept
* When you are prompted with login as, enter: ec2-user.

macOS & Linux users

* Choose the [Details] & then select [show]
* Choose the Download PEM button & save the labuser.pem file.
* Note the On Prem Linux Instance address, if it is displayed
* Exit the Details panel by Choosing the x
* Open a terminal window & change directory

  cd ~/Downloads.

* Run the following command (<public-ip>, replace this with On Prem Linux Instance.)

  'ssh -i labuser.pem ec2-user@<public-ip>

you should now be Connected to the instance

Task 6: verifying that the data is migrated

* In the Services Search box, search for and choose S3 to open the S3 console.
* select the bucket that you created in the US East (Ohio) Region.
  - verify that the 20 image files are listed
* Return to the Buckets page and select the bucket that you created in the US West (Oregon) Region.

Submitting your work

* Choose [submit] and the Click on [End Lab]