



7/29/2022

# Project-01

Configure more than one webserver with proper load balancing, auto scaling and SSL configuration.



Sumit Mishra  
SIC: 190310286

# 1. Creating and configuring a VPC.

## 1. Create a VPC

| Name         | VPC ID                | State     | IPv4 CIDR  | IPv6 CIDR | DHCP   |
|--------------|-----------------------|-----------|------------|-----------|--------|
| personal-vpc | vpc-083ef3fe083e0f251 | Available | 7.0.0.0/16 | -         | dopt-0 |

## 2. Create a subnet inside the VPC to host servers.

| subnet-0c89d2c7eab4ba5ac / web-subnet       |   |   |                                  | Actions ▾ |
|---|---|---|----------------------------------|-----------|
| Details                                     |   |   |                                  |           |
| Subnet ID<br>subnet-0c89d2c7eab4ba5ac       | Subnet ARN<br>arn:aws:ec2:ap-south-1:258046353232:subnet/subnet-0c89d2c7eab4ba5ac | State<br>Available                            | IPv4 CIDR<br>7.0.1.0/24          |           |
| Available IPv4 addresses<br>247             | IPv6 CIDR<br>-  | Availability Zone<br>ap-south-1a              | Availability Zone ID<br>aps1-az1 |           |
| VPC<br>vpc-083ef3fe083e0f251   personal-vpc | Route table<br>rtb-0b7cda26fa0da090d   web-server-rt                              | Network ACL<br>acl-000ec8cc218e077b7          | Default subnet<br>No             |           |
| Auto-assign public IPv4 address<br>Yes      | Auto-assign IPv6 address<br>No  | Auto-assign customer-owned IPv4 address<br>No | Customer-owned IPv4 pool<br>-    |           |
| Outpost ID<br>-                             | IPv4 CIDR reservations<br>-   | IPv6 CIDR reservations<br>-                   | IPv6-only<br>No                  |           |
| Hostname type<br>IP name                    | Resource name DNS A record<br>Disabled  | Resource name DNS AAAA record<br>Disabled     | DNS64<br>Disabled                |           |
| Owner<br>258046353232                       |   |   |                                  |           |

## 3. Created an Internet Gateway for the servers to access the internet through it.

|  |                   |  |                       |
|--|-------------------|--|-----------------------|
| VPC  | Internet gateways | igw-0f10d2feb4472a074                          | Actions ▾             |
| igw-0f10d2feb4472a074 / personal-igw         |                   |  |                       |
| Details Info                                 |                   |  |                       |
| Internet gateway ID<br>igw-0f10d2feb4472a074 | State<br>Attached | VPC ID<br>vpc-083ef3fe083e0f251   personal-vpc | Owner<br>258046353232 |
| Tags   |                   |  |                       |
| <input type="text"/> Search tags             |                   |  |                       |
| Key  | Value             | Manage tags                                    |                       |
| Name   | personal-igw      | < 1 > <input type="button"/>                   |                       |

#### 4. Created and configured the route table for the subnet created.

The screenshot shows the AWS Route Tables console for a VPC. The route table ID is rtb-0b7cda26fa0da090d, and it is associated with the subnet subnet-0c89d2c7eab4ba5ac / web-subnet. The table has one explicit subnet association and no edge associations. It contains six routes, all of which are active and propagated. The routes target various AWS services like igw-0f10d2feb4472a074 and local.

| Destination       | Target                | Status | Propagated |
|-------------------|-----------------------|--------|------------|
| 0.0.0.0/0         | igw-0f10d2feb4472a074 | Active | No         |
| 7.0.0.0/16        | local                 | Active | No         |
| 13.232.122.87/32  | igw-0f10d2feb4472a074 | Active | No         |
| 13.233.197.113/32 | igw-0f10d2feb4472a074 | Active | No         |
| 43.205.130.90/32  | igw-0f10d2feb4472a074 | Active | No         |
| 65.0.95.54/32     | igw-0f10d2feb4472a074 | Active | No         |

## 2. Configuring Web servers, domain, hosting.

### 1. Launching 4 EC2 instances with Linux AMI.

The screenshot shows the AWS Instances console listing four EC2 instances. All instances are running and belong to the t2.micro instance type. They are located in the ap-south-1a availability zone and have public IPv4 addresses assigned to them.

| Name       | Instance ID         | Instance state | Instance type | Status check      | Alarm status | Availability Zone | Public IPv4 DNS | Public IPv4 ... |
|------------|---------------------|----------------|---------------|-------------------|--------------|-------------------|-----------------|-----------------|
| webserver4 | i-0c31d18055bbbc84d | Running        | t2.micro      | 2/2 checks passed | No alarms    | ap-south-1a       | -               | 3.110.18.77     |
| webserver3 | i-0ca0d578d9ae4db13 | Running        | t2.micro      | 2/2 checks passed | No alarms    | ap-south-1a       | -               | 13.235.92.242   |
| webserver1 | i-01ef2264909f51ee6 | Running        | t2.micro      | 2/2 checks passed | No alarms    | ap-south-1a       | -               | 43.205.77.193   |
| webserver2 | i-0d6fe1c4744994690 | Running        | t2.micro      | 2/2 checks passed | No alarms    | ap-south-1a       | -               | 43.205.53.124   |

### 2. Assigning Elastic IP addresses to all the four servers.

The screenshot shows the AWS Instances console after assigning Elastic IP addresses. The instances now have both a public IPv4 address and an assigned Elastic IP address. The Elastic IP column lists the assigned static IP addresses for each instance.

| Name       | Instance ID         | Instance state | Instance type | Status check      | Public IPv4 ... | Elastic IP    |
|------------|---------------------|----------------|---------------|-------------------|-----------------|---------------|
| webserver4 | i-0c31d18055bbbc84d | Running        | t2.micro      | 2/2 checks passed | 3.110.18.77     | 3.110.18.77   |
| webserver3 | i-0ca0d578d9ae4db13 | Running        | t2.micro      | 2/2 checks passed | 13.235.92.242   | 13.235.92.242 |
| webserver1 | i-01ef2264909f51ee6 | Running        | t2.micro      | 2/2 checks passed | 43.205.77.193   | 43.205.77.193 |
| webserver2 | i-0d6fe1c4744994690 | Running        | t2.micro      | 2/2 checks passed | 43.205.53.124   | 43.205.53.124 |

3. Creating a Hosted zone in AWS Route 53 and adding hosting records for root domain(sumitmishra.info) and sub-domain(www.sumitmishra.info).

Route 53 > Hosted zones > sumitmishra.info

**sumitmishra.info** [Info](#)

**Hosted zone details** [Edit hosted zone](#)

**Records (4)** [Info](#) [Delete zone](#) [Test record](#) [Configure query logging](#)

**Records (4)** [Info](#) [Delete record](#) [Import zone file](#) [Create record](#)

Automatic mode is the current search behavior optimized for best filter results. To change modes go to settings.

| <input type="checkbox"/> | Record name          | Type | Routine... | Differ... | Value/Route traffic to  |
|--------------------------|----------------------|------|------------|-----------|---|
| <input type="checkbox"/> | sumitmishra.info     | A    | Simple     | -         | 43.205.77.193   |
| <input type="checkbox"/> | sumitmishra.info     | NS   | Simple     | -         | ns-15.awsdns-01.com.<br>ns-1594.awsdns-07.co.uk.<br>ns-985.awsdns-59.net.<br>ns-1149.awsdns-15.org. |
| <input type="checkbox"/> | sumitmishra.info     | SOA  | Simple     | -         | ns-15.awsdns-01.com. awsdns-hostmaster.amazon.com. 1 7200 900 1209600 86400                         |
| <input type="checkbox"/> | www.sumitmishra.info | A    | Simple     | -         | 43.205.77.193   |

4. Replacing Nameservers in GoDaddy Domain (sumitmishra.info) with AWS Route 53 Nameservers.

**Nameservers**

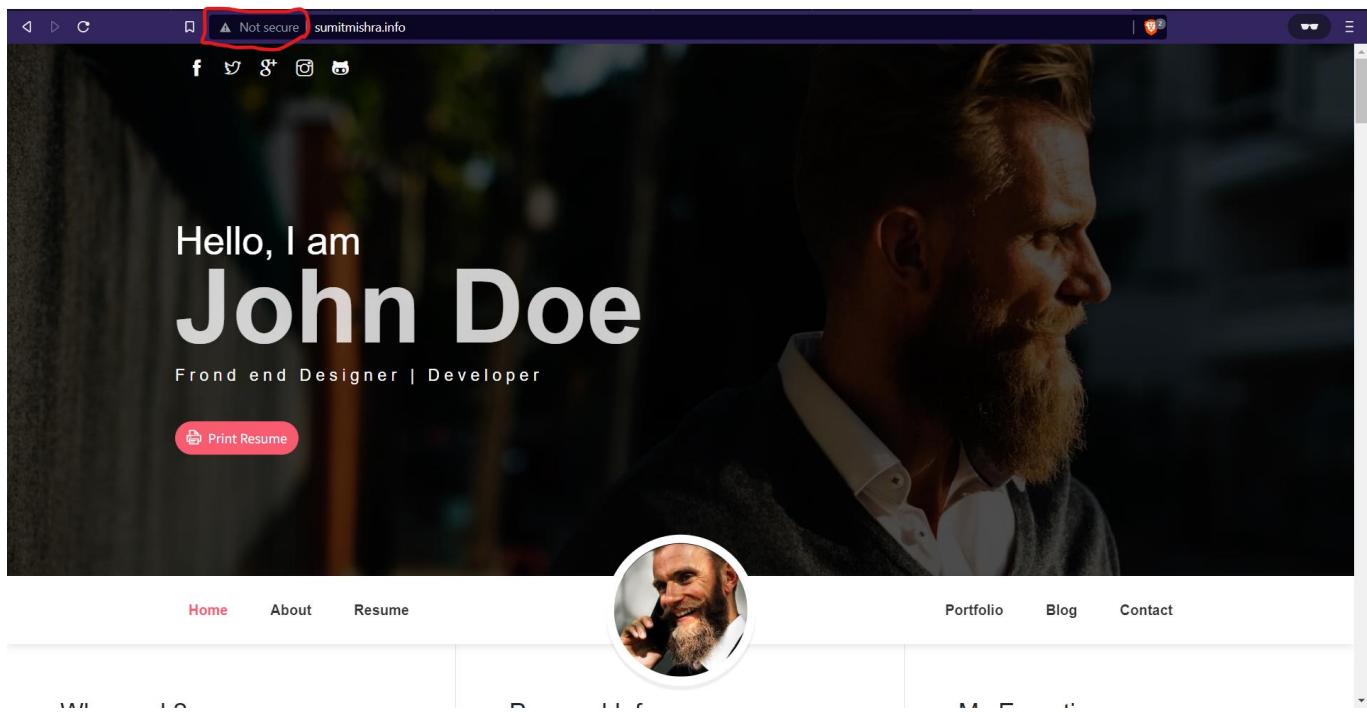
Using custom nameservers [Change](#)

**Nameservers** [?](#)

- ns-15.awsdns-01.com
- ns-1594.awsdns-07.co.uk
- ns-985.awsdns-59.net
- ns-1149.awsdns-15.org

3. Configuring SSL/TLS certificate for a webserver using let us encrypt Apache certbot.

1. Opening the website hosted in webserver one without SSL/TLS certificate we get the not secured warning in the left most side of URL bar.



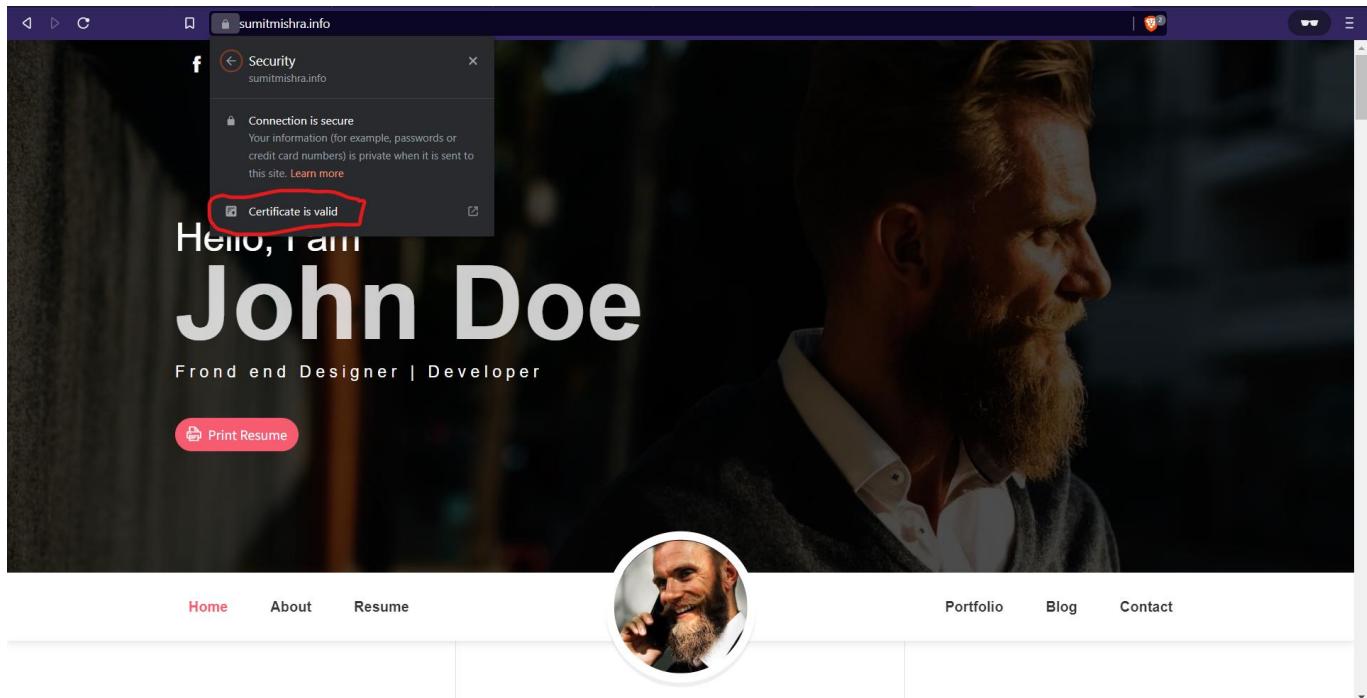
## 2. Now configuring the SSL certificate through let's encrypt using ssh client.

```
[root@ip-7-0-1-148 ~]# certbot --apache
Saving debug log to /var/log/letsencrypt/letsencrypt.log
Plugins selected: Authenticator apache, Installer apache

Which names would you like to activate HTTPS for?
-----
1: sumitmishra.info
2: www.sumitmishra.info
-----
Select the appropriate numbers separated by commas and/or spaces, or leave input
blank to select all options shown (Enter 'c' to cancel):
Requesting a certificate for sumitmishra.info and www.sumitmishra.info
Performing the following challenges:
http-01 challenge for sumitmishra.info
http-01 challenge for www.sumitmishra.info
Waiting for verification...
Cleaning up challenges
Created an SSL vhost at /etc/httpd/conf.d/vhost-le-ssl.conf
Deploying Certificate to VirtualHost /etc/httpd/conf.d/vhost-le-ssl.conf
Deploying Certificate to VirtualHost /etc/httpd/conf.d/vhost-le-ssl.conf
Redirecting vhost in /etc/httpd/conf.d/vhost.conf to ssl vhost in /etc/httpd/conf.d/vhost-le-ssl.conf

-----
Congratulations! You have successfully enabled https://sumitmishra.info and
https://www.sumitmishra.info
-----
```

## 3. Now again hitting the URL sumitmishra.info and www.sumitmishra.info to see that the website shows secured in the leftmost area of the URL bar, this means that SSL certificate has been enabled.



4. Creating a target group using the four servers and configuring load balancing along with SSL certificate using let's encrypt for the load balancer.

## 1. Creating a target group.

EC2 > Target groups

| Target groups (1/1) <a href="#">Info</a> |                 |   |      |          |             |                 | <a href="#">Actions</a> | <a href="#">Create target group</a> |
|--|-----------------|---|------|----------|-------------|-----------------|-------------------------|-------------------------------------|
| <input checked="" type="checkbox"/>      | Name            | ARN   | Port | Protocol | Target type | Load balancer   | VPC ID                  |                                     |
| <input checked="" type="checkbox"/>      | personal-vpc-tg | arn:aws:elasticloadbalancing:ap-south-1:123456789012:targetgroup/personal-vpc-tg/1234567890123456 | 80   | HTTP     | Instance    | None associated | vpc-083ef3fe083e0f25    |                                     |

**Target group: personal-vpc-tg**

| Registered targets (4)   |                     |            |      |             |                       |  | <a href="#">Actions</a> | <a href="#">Deregister</a> | <a href="#">Register targets</a> |
|--------------------------|---------------------|------------|------|-------------|-----------------------|--|-------------------------|----------------------------|----------------------------------|
| <input type="checkbox"/> | Instance ID         | Name       | Port | Zone        | Health status         | Health status details  |                         |                            |                                  |
| <input type="checkbox"/> | i-0c30c9909807edd2d | webserver1 | 80   | ap-south-1a | <span>🟡</span> unused | Target group is not configured to receive traffic from the load balancer |                         |                            |                                  |
| <input type="checkbox"/> | i-02640c5161f8a7117 | webserver4 | 80   | ap-south-1b | <span>🟡</span> unused | Target group is not configured to receive traffic from the load balancer |                         |                            |                                  |
| <input type="checkbox"/> | i-0bbe0940452be4101 | webserver2 | 80   | ap-south-1b | <span>🟡</span> unused | Target group is not configured to receive traffic from the load balancer |                         |                            |                                  |
| <input type="checkbox"/> | i-0e0945972773e248c | webserver3 | 80   | ap-south-1a | <span>🟡</span> unused | Target group is not configured to receive traffic from the load balancer |                         |                            |                                  |

## 2. Creating a load balancer.

## Basic configuration

### Load balancer name

Name must be unique within your AWS account and cannot be changed after the load balancer is created.

A maximum of 32 alphanumeric characters including hyphens are allowed, but the name must not begin or end with a hyphen.

### Scheme [Info](#)

Scheme cannot be changed after the load balancer is created.

**Internet-facing**

An internet-facing load balancer routes requests from clients over the internet to targets. Requires a public subnet. [Learn more](#) 

**Internal**

An internal load balancer routes requests from clients to targets using private IP addresses.

### IP address type [Info](#)

Select the type of IP addresses that your subnets use.

**IPv4**

Recommended for internal load balancers.

**Dualstack**

Includes IPv4 and IPv6 addresses.

## Network mapping [Info](#)

The load balancer routes traffic to targets in the selected subnets, and in accordance with your IP address settings.

### VPC [Info](#)

Select the virtual private cloud (VPC) for your targets. Only VPCs with an internet gateway are enabled for selection. The selected VPC cannot be changed after the load balancer is created. To confirm the VPC for your targets, view your target groups .

  
vpc-083ef3fe083e0f251  
IPv4: 7.0.0.0/16

### Mappings [Info](#)

Select at least one Availability Zone and one subnet for each zone. We recommend selecting at least two Availability Zones. The load balancer will route traffic only to targets in the selected Availability Zones. Zones that are not supported by the load balancer or VPC cannot be selected. Subnets can be added, but not removed, once a load balancer is created.

ap-south-1a

#### Subnet

 web-subnet-1 

#### IPv4 settings

Assigned by AWS

ap-south-1b

#### Subnet

 web-subnet-2 

#### IPv4 settings

Assigned by AWS

**sg-05a720fd534a6ba81 - personal-vpc-lb-sg****Details**

|  |   |   |                                 |
|--|---|---|---------------------------------|
| Security group name<br>sg personal-vpc-lb-sg | Security group ID<br>sg-05a720fd534a6ba81   | Description<br>security group for personal VPC subnet Load balancer | VPC ID<br>vpc-083ef3fe083e0f251 |
| Owner<br>258046353232                        | Inbound rules count<br>2 Permission entries | Outbound rules count<br>1 Permission entry                          |                                 |

**Inbound rules**   **Outbound rules**   **Tags****Inbound rules (2)**

| Inbound rules (2)  |      |                        |            |       |          |            |           |             |  |  |
|--|------|------------------------|------------|-------|----------|------------|-----------|-------------|--|--|
| <input type="button" value="C"/> Manage tags <input type="button" value="Edit inbound rules"/> |      |                        |            |       |          |            |           |             |  |  |
|  | Name | Security group rule... | IP version | Type  | Protocol | Port range | Source    | Description |  |  |
| <input type="checkbox"/>   | -    | sgr-0515a0ed07dcd40... | IPv4       | HTTP  | TCP      | 80         | 0.0.0.0/0 | -           |  |  |
| <input type="checkbox"/>   | -    | sgr-06335d143f137fa86  | IPv4       | HTTPS | TCP      | 443        | 0.0.0.0/0 | -           |  |  |

| Load balancers  |  |              |                       |                          |             |                                 |                                     |
|---|--|--------------|-----------------------|--------------------------|-------------|---------------------------------|-------------------------------------|
| <input type="text" value="personal-vpc-lb"/> Add filter |  |              |                       |                          |             |                                 |                                     |
| Name  | DNS name   | State        | VPC ID                | Availability Zones       | Type        | Created At                      | Monitoring                          |
| personal-vpc-lb   | personal-vpc-lb-845205905.ap-south-1.elb.amazonaws.com | Provisioning | vpc-083ef3fe083e0f251 | ap-south-1b, ap-south-1a | application | July 30, 2022 at 9:39:23 AM ... | <input type="button" value="Edit"/> |

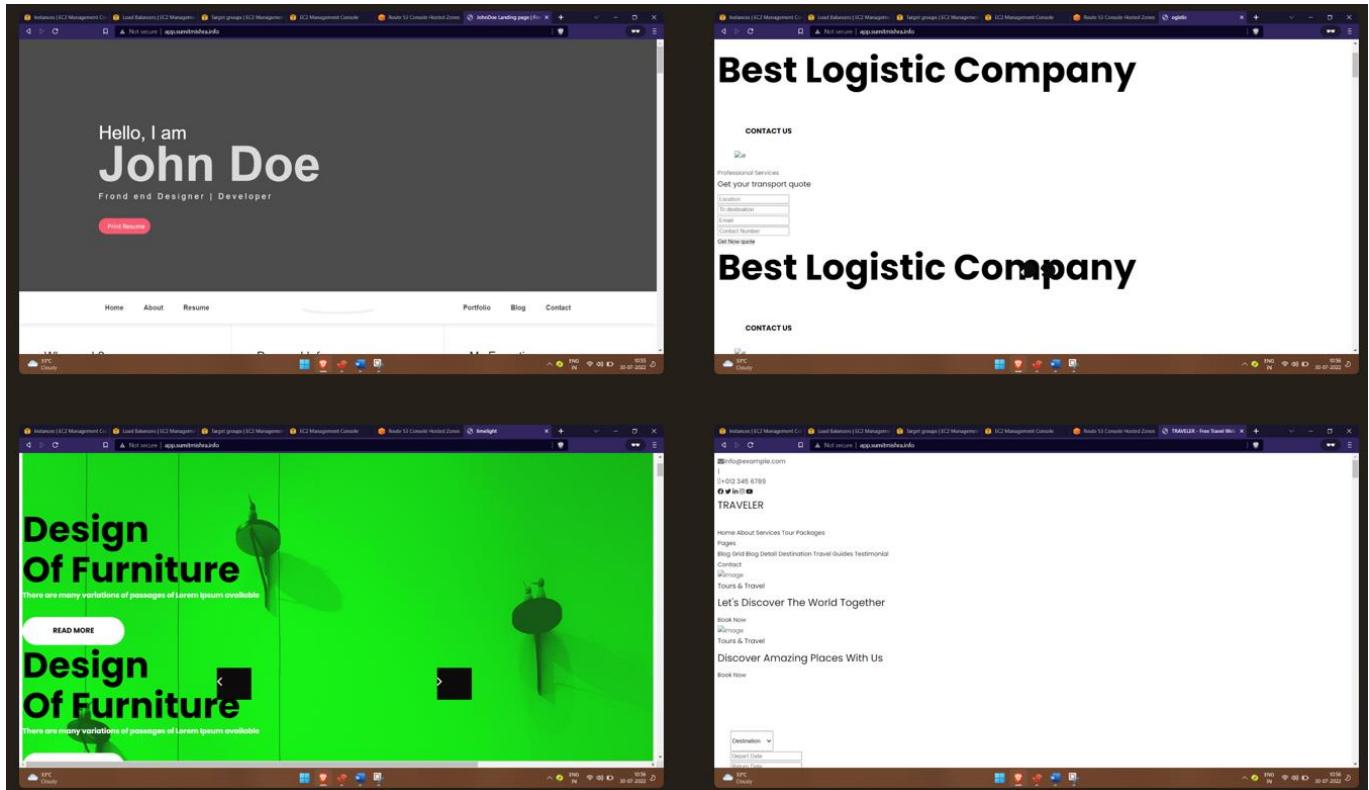
| Load balancer: personal-vpc-lb  |  | Description | Listeners | Monitoring | Integrated services | Tags |
|---|--|-------------|-----------|------------|---------------------|------|
| <b>Basic Configuration</b>  |  |             |           |            |                     |      |
| <b>Name</b> personal-vpc-lb<br><b>ARN</b> arn:aws:elasticloadbalancing:ap-south-1:258046353232:loadbalancer/app/personal-vpc-lb/03148219436b89bf<br><b>DNS name</b> personal-vpc-lb-845205905.ap-south-1.elb.amazonaws.com (A Record)<br><b>State</b> Provisioning<br><b>Type</b> application<br><b>Scheme</b> internet-facing<br><b>IP address type</b> ipv4<br><a href="#">Edit IP address type</a><br><b>VPC</b> vpc-083ef3fe083e0f251<br><b>Availability Zones</b> subnet-0225018fd4692efec - ap-south-1b<br>IPv4 address: Assigned by AWS<br>subnet-0c89d2c7eab4ba5ac - ap-south-1a<br>IPv4 address: Assigned by AWS |  |             |           |            |                     |      |
|   |  |             |           |            |                     |      |
|   |  |             |           |            |                     |      |

### 3. Creating a new sub-domain and associating the load balancer's DNS to that domain name.

**Record details**

|   |  |                    |
|---|--|--------------------|
| <input type="button" value="Edit record"/>  | Record name<br><input type="text" value="app.sumitmishra.info"/> | Record type<br>A   |
| Value<br><input type="text" value="dualstack.personal-vpc-lb-845205905.ap-south-1.elb.amazonaws.com."/> | Alias<br>Yes   | TTL (seconds)<br>- |
| Routing policy<br>Simple  |  |                    |
|   |  |                    |

### 4. Server is distributing the traffic, but SSL certificate has not been configured.



## 5. SSL configuration using AWS Certificate Manager.

- Request a certificate.

AWS Certificate Manager > Certificates > Request certificate > Request public certificate

### Request public certificate

**Domain names**

Fully qualified domain name [Info](#)

Remove
  

Remove

Add another name to this certificate

You can add additional names to this certificate. For example, if you're requesting a certificate for "www.example.com", you might want to add the name "example.com" so that customers can reach your site by either name.

**Select validation method [Info](#)**

Select a method for validating domain ownership

DNS validation - recommended  
Choose this option if you are authorized to modify the DNS configuration for the domains in your certificate request.

Email validation  
Choose this option if you do not have permission or cannot obtain permission to modify the DNS configuration for the domains in your certificate request.

**Tags [Info](#)**

To help you manage your certificates you can optionally assign your own metadata to each resource in the form of tags.

|  |   |
|--|---|
| Tag key                                | Tag value - optional                      |
| <input type="text" value="Name"/>      | <input type="text" value="personal-SSL"/> |
| <input type="button" value="Add tag"/> | <input type="button" value="Remove tag"/> |

You can add 49 more tag(s).

Cancel Previous Request

- After requesting, add the records to the route 53 table.

## Domains (2)

[Create records in Route 53](#)[Export to CSV](#)

&lt; 1 &gt;

| Domain             | Status               | Renewal status | Type  | CNAME name  | CNAME value   |
|--------------------|----------------------|----------------|-------|---|---|
| sumitmishra.info   | <span>Success</span> | -              | CNAME | _9ef0869ebcb98a8cef7a1082a6280e1.sumitmishra.info | _d9b8e5f79784562a53896b12c2adeb95.vrtfgqhx.acm-validations.aws. |
| *.sumitmishra.info | <span>Success</span> | -              | CNAME | _9ef0869ebcb98a8cef7a1082a6280e1.sumitmishra.info | _d9b8e5f79784562a53896b12c2adeb95.vrtfgqhx.acm-validations.aws. |

- c. After certificate has been issued, go to load balancer -> listener -> add new listener -> select https and forward it to target group created and select the certificate as the created one.

arn:aws:elasticloadbalancing:ap-south-1:258046353232:listener/app/personal-vpc-lb/03148219436b89bf/aa0d128763...

### Listener details

A listener is a process that checks for connection requests, using the protocol and port you configure. Traffic received by the listener is then routed per your specification. You can specify multiple rules and multiple certificates per listener after the load balancer is created.

| Protocol | Port             |
|----------|------------------|
| HTTPS    | : 443<br>1-65535 |

**Default actions** [Info](#)  
Specify the default actions for traffic on this listener. Default actions apply to traffic that does not meet the conditions of rules on your listener. Rules can be configured after the listener is created.

**1. Forward to** [Info](#) [Remove](#)

| Target group                                   | Weight (0-999) |
|--|----------------|
| personal-vpc-tg<br>Target type: Instance, IPv4 | HTTP 1 100%    |
| Select a target group                          | 0              |

**Enable group-level stickiness** [Info](#)  
If you enable stickiness for your target group, requests routed to it remain in the same group for the duration you specify.

[Add action](#)

### Secure listener settings

**Security policy**  
Your load balancer uses a Secure Socket Layer (SSL) negotiation configuration, known as a security policy, to negotiate SSL connections with clients.

ELBSecurityPolicy-2016-08

[Compare security policies](#)

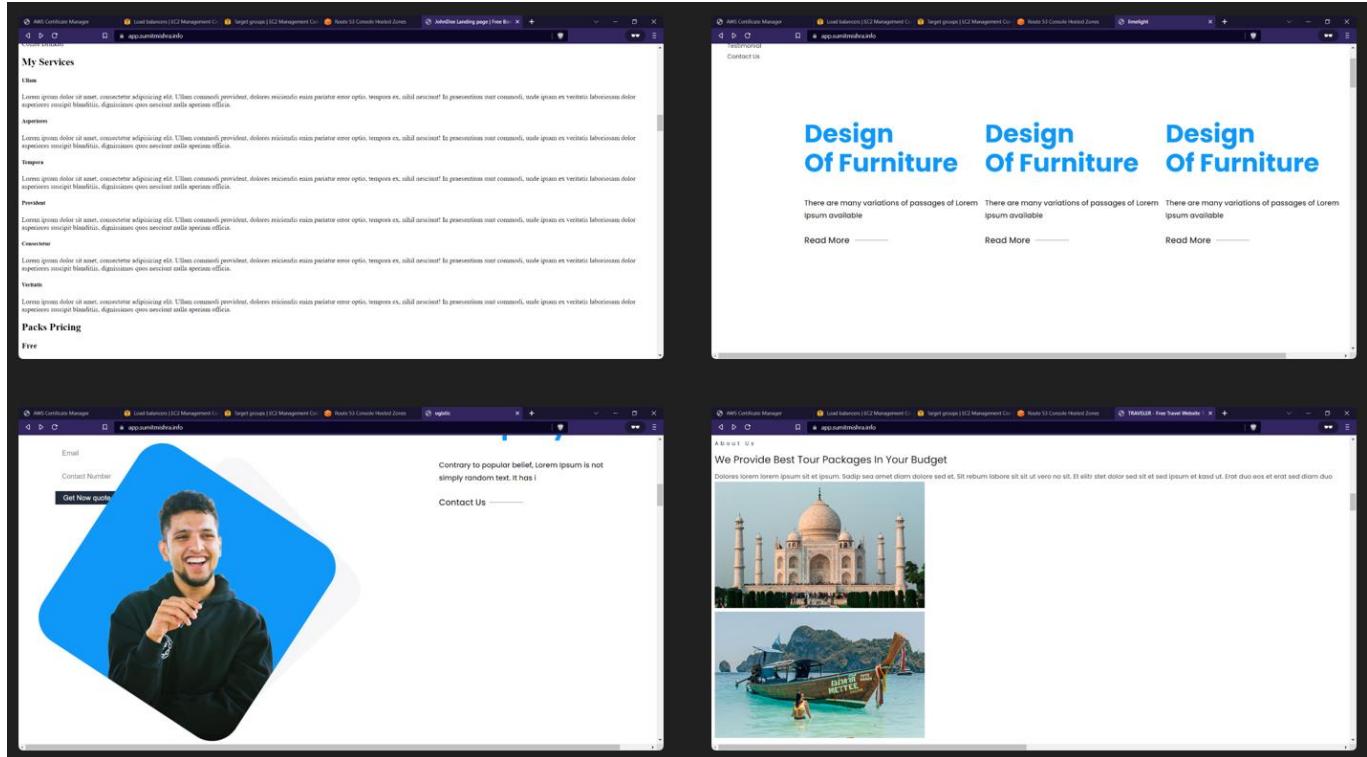
**Default SSL/TLS certificate**  
The certificate used if a client connects without SNI protocol, or if there are no matching certificates. This certificate will automatically be added to your listener certificate list.

From ACM sumitmishra.info b5b6c55d-9acd-4417-9390-ce56a147794a [Create certificate](#)

[Request new ACM certificate](#)

[Cancel](#) [Save changes](#)

6. SSL has been successfully enabled if we hit the URL with <https://app.sumitmishra.info> and load is balanced between the backend web servers.



## 5. Configuring auto scaling to an instance.

### 1. Create an Image of webserver1 EC2 instance.

Screenshot of the AWS Management Console showing the Instances page. The 'Instances (1/4) Info' table lists four instances: webserver1 (Stopped, t2.micro), webserver3 (Stopped, t2.micro), webserver2 (Stopped, t2.micro), and webserver4 (Stopped, t2.micro). A context menu is open over the first instance, showing options like 'Create image', 'Create template from instance', and 'Launch more like this'. The 'Actions' dropdown menu on the right includes 'Public IPv4 DNS' and 'Image and templates'.

Screenshot of the AWS Management Console showing the Amazon Machine Images (AMIs) page. It displays a single AMI entry: 'Owned by me' (ami-0ced563eea51eed5d), 'AMI name' (webserver1-img), 'Source' (258046353232/webserver1-img), 'Owner' (258046353232), 'Visibility' (Private), and 'Status' (Available). Buttons for 'Recycle Bin', 'EC2 Image Builder', and 'Launch instance from AMI' are visible.

### 2. Create a new Load balancer “webserver1-lb” specifically for that server and assign a new configured target group for the “webserver1-tg” not having any instance attached to it.

**Details**

arn:aws:elasticloadbalancing:ap-south-1:258046353232:targetgroup/webserver1-tg/61b0e35afe4ebf98

|                         |  |   |   |
|-------------------------|--|---|---|
| Target type<br>Instance | Protocol : Port<br>HTTP: 80                      | Protocol version<br>HTTP1                       | VPC<br><a href="#">vpc-083ef3fe083e0f251</a>    |
| IP address type<br>IPv4 | Load balancer<br><a href="#">None associated</a> |   |   |
| Total targets<br>0      | Healthy<br><span style="color: green;">0</span>  | Unhealthy<br><span style="color: red;">0</span> | Unused<br><span style="color: gray;">0</span>   |
|                         |  |   | Initial<br><span style="color: gray;">0</span>  |
|                         |  |   | Draining<br><span style="color: gray;">0</span> |

**Targets** | Monitoring | Health checks | Attributes | Tags

**Registered targets (0)**

Filter resources by property or value

C Deregister Register targets

Instance ID Name Port Zone Health status Health status details

No registered targets

Register targets

- Now, created a new launch configuration using image of the “webserver1” instance and also checking on the option of detailed monitoring using AWS cloudwatch.

EC2 > Launch configurations > Create launch configuration

### Create launch configuration Info

**Launch configuration name**

Name

**Amazon machine image (AMI) Info**

AMI

**Instance type Info**

Instance type  
 Choose instance type

## Additional configuration - optional

Purchasing option [Info](#)

Request Spot Instances

IAM instance profile [Info](#)

Select IAM role ▾

Monitoring [Info](#)

Enable EC2 instance detailed monitoring within CloudWatch

EBS-optimized instance

Launch as EBS-optimized instance

### ► Advanced details

i Later, if you want to use a different launch configuration, you can create a new one and apply it to any Auto Scaling group. Existing launch configurations cannot be edited.

## Storage (volumes) [Info](#)

### EBS volumes

[Remove](#) ▾

| <input type="checkbox"/> | Volume type | Devices   | Snapshot               | Size (GiB) | Volume type             |
|--------------------------|-------------|-----------|------------------------|------------|-------------------------|
| <input type="checkbox"/> | Root        | /dev/xvda | snap-006011a3351499f7d | 8          | General purpose SSD (g) |

[+ Add new volume](#)

i Free tier eligible customers can get up to 30 GB of EBS storage. [Learn more](#) about free usage tier eligibility and usage restrictions.

## Security groups [Info](#)

Assign a security group

- Create a new security group
- Select an existing security group

### Security groups

[Copy to new](#)

[View rules](#)

◀ 1 ▶

| <input type="checkbox"/>            | Security group ID    | Name  | VPC ID                | Description  |
|-------------------------------------|----------------------|---|-----------------------|--|
| <input type="checkbox"/>            | sg-05a720fd534a6ba81 | personal-vpc-lb-sg                            | vpc-083ef3fe083e0f251 | security group for personal VPC subnet Load balancer   |
| <input type="checkbox"/>            | sg-072e8fa63c711a4ff | OpenVPN Access Server-2.8.5-AutogenByAWSMP--1 | vpc-01d987e17c3785149 | This security group was generated by AWS Marketplace and is based on recommended settings for OpenVPN Access Server version 2.8.5 provided by OpenVPN Inc. |
| <input type="checkbox"/>            | sg-08c979ce29bbae3f5 | default                                       | vpc-083ef3fe083e0f251 | default VPC security group   |
| <input checked="" type="checkbox"/> | sg-0bd36c936a052ff4e | launch-wizard-3                               | vpc-083ef3fe083e0f251 | launch-wizard-3 created 2022-07-29T09:48:41.971Z   |
| <input type="checkbox"/>            | sg-0e9f82d5ed4d73a2d | default                                       | vpc-01d987e17c3785149 | default VPC security group   |

⚠ Rules with source of 0.0.0.0/0 allow all IP addresses to access your instance. We recommend setting security group rules to allow access from known IP addresses only.

**Key pair (login) [Info](#)**

Key pair options

Choose an existing key pair ▾

Existing key pair

VPC-server-sumit ▾

I acknowledge that I have access to the selected private key file (VPC-server-sumit.pem), and that without this file, I won't be able to log into my instance.

[Cancel](#) [Create launch configuration](#)

#### 4. Created an auto scaling group using launch dynamic configuration by providing the created launch configuration.

Step 1  
**Choose launch template or configuration**

Step 2  
Choose instance launch options

Step 3 (optional)  
Configure advanced options

Step 4 (optional)  
Configure group size and scaling policies

Step 5 (optional)  
Add notifications

Step 6 (optional)  
Add tags

Step 7  
Review

**Choose launch template or configuration [Info](#)**

Specify a launch template that contains settings common to all EC2 instances that are launched by this Auto Scaling group. If you currently use launch configurations, you might consider migrating to launch templates.

**Name**

Auto Scaling group name  
Enter a name to identify the group.

webserver1-asg

Must be unique to this account in the current Region and no more than 255 characters.

**Launch configuration [Info](#)** [Switch to launch template](#)

Launch configuration  
Choose a launch configuration that contains the instance-level settings, such as the Amazon Machine Image (AMI), instance type, key pair, and security groups.

| Launch configuration | AMI ID                | Date created   |
|----------------------|-----------------------|--|
| webserver1-lc        | ami-0ced563eea51eed5d | Mon Aug 01 2022 10:51:52<br>GMT+0530 (India Standard Time) |
| Security groups      | Instance type         | Key pair name  |
| sg-0bd36c936a052ff4e | t2.micro              | VPC-server-sumit   |

[Cancel](#) [Next](#)

Step 1  
Choose launch template or configuration

---

Step 2  
Choose instance launch options

---

Step 3 (optional)  
Configure advanced options

---

Step 4 (optional)  
Configure group size and scaling policies

---

Step 5 (optional)  
Add notifications

---

Step 6 (optional)  
Add tags

---

Step 7  
Review

## Choose instance launch options Info

Choose the VPC network environment that your instances are launched into, and customize the instance types and purchase options.

### Network Info

For most applications, you can use multiple Availability Zones and let EC2 Auto Scaling balance your instances across the zones. The default VPC and default subnets are suitable for getting started quickly.

#### VPC

Choose the VPC that defines the virtual network for your Auto Scaling group.

vpc-083ef3fe083e0f251 (personal-vpc) ▾



[Create a VPC](#)

#### Availability Zones and subnets

Define which Availability Zones and subnets your Auto Scaling group can use in the chosen VPC.

Select Availability Zones and subnets ▾



ap-south-1a | subnet-0c89d2c7eab4ba5ac (web-subnet-1)   
7.0.1.0/24

ap-south-1b | subnet-0225018fd4692efec (web-subnet-2)   
7.0.2.0/24

[Create a subnet](#)

[Cancel](#)

[Previous](#)

[Skip to review](#)

[Next](#)

[EC2](#) > [Auto Scaling groups](#) > Create Auto Scaling group

Step 1  
Choose launch template or configuration

---

Step 2  
Choose instance launch options

---

Step 3 (optional)  
Configure advanced options

---

Step 4 (optional)  
Configure group size and scaling policies

---

Step 5 (optional)  
Add notifications

---

Step 6 (optional)  
Add tags

---

Step 7  
Review

## Configure advanced options Info

Choose a load balancer to distribute incoming traffic for your application across instances to make it more reliable and easily scalable. You can also set options that give you more control over health check replacements and monitoring.

### Load balancing - optional Info

Use the options below to attach your Auto Scaling group to an existing load balancer, or to a new load balancer that you define.



No load balancer  
Traffic to your Auto Scaling group will not be fronted by a load balancer.



Attach to an existing load balancer  
Choose from your existing load balancers.



Attach to a new load balancer  
Quickly create a basic load balancer to attach to your Auto Scaling group.

### Attach to an existing load balancer

Select the load balancers that you want to attach to your Auto Scaling group.



Choose from your load balancer target groups  
This option allows you to attach Application, Network, or Gateway Load Balancers.



Choose from Classic Load Balancers

**Existing load balancer target groups**  
Only instance target groups that belong to the same VPC as your Auto Scaling group are available for selection.

Select target groups ▾ C

webserver1-tg | HTTP X  
Application Load Balancer: webserver1-lb

### Health checks - optional

Health check type [Info](#)  
EC2 Auto Scaling automatically replaces instances that fail health checks. If you enabled load balancing, you can enable ELB health checks in addition to the EC2 health checks that are always enabled.

EC2  ELB

Health check grace period  
The amount of time until EC2 Auto Scaling performs the first health check on new instances after they are put into service.

20 seconds

### Additional settings - optional

Monitoring [Info](#)

Enable group metrics collection within CloudWatch

Default instance warmup [Info](#)  
The amount of time that CloudWatch metrics for new instances do not contribute to the group's aggregated instance metrics, as their usage data is not reliable yet.

Enable default instance warmup

[Cancel](#) [Previous](#) [Skip to review](#) [Next](#)

EC2 > Auto Scaling groups > Create Auto Scaling group

Step 1  
Choose launch template or configuration

Step 2  
Choose instance launch options

Step 3 (optional)  
Configure advanced options

Step 4 (optional)  
Configure group size and scaling policies

Step 5 (optional)  
Add notifications

Step 6 (optional)  
Add tags

Step 7  
Review

### Configure group size and scaling policies [Info](#)

Set the desired, minimum, and maximum capacity of your Auto Scaling group. You can optionally add a scaling policy to dynamically scale the number of instances in the group.

#### Group size - optional [Info](#)

Specify the size of the Auto Scaling group by changing the desired capacity. You can also specify minimum and maximum capacity limits. Your desired capacity must be within the limit range.

Desired capacity  
1

Minimum capacity  
1

Maximum capacity  
3

#### Scaling policies - optional

Choose whether to use a scaling policy to dynamically resize your Auto Scaling group to meet changes in demand. [Info](#)

Target tracking scaling policy  
Choose a desired outcome and leave it to the scaling policy to add and remove capacity as needed to achieve that outcome.

None

#### Instance scale-in protection - optional

Instance scale-in protection  
If protect from scale in is enabled, newly launched instances will be protected from scale in by default.

Enable instance scale-in protection

[Cancel](#) [Previous](#) [Skip to review](#) [Next](#)

Step 1  
Choose launch template or configuration

Step 2  
Choose instance launch options

Step 3 (optional)  
Configure advanced options

Step 4 (optional)  
Configure group size and scaling policies

Step 5 (optional)  
Add notifications

Step 6 (optional)  
Add tags

Step 7  
Review

### Add tags Info

Add tags to help you search, filter, and track your Auto Scaling group across AWS. You can also choose to automatically add these tags to instances when they are launched.

ⓘ You can optionally choose to add tags to instances (and their attached EBS volumes) by specifying tags in your launch template. We recommend caution, however, because the tag values for instances from your launch template will be overridden if there are any duplicate keys specified for the Auto Scaling group. X

| Tags (1)                |                   |                                     |
|-------------------------|-------------------|-------------------------------------|
| Key                     | Value - optional  | Tag new instances                   |
| Name                    | webserver1-scaled | <input checked="" type="checkbox"/> |
| <a href="#">Add tag</a> |                   | <a href="#">Remove</a>              |
| 49 remaining            |                   |                                     |

[Cancel](#) [Previous](#) [Next](#)

When auto scaling group will be created then by default an instance using the image created will be launched in the backend.

**Instances (2) Info**

| <input type="checkbox"/> | Name             | Instance ID         | Instance state       | Instance type | Status check                   | Alarm status | Availability Zone |
|--------------------------|------------------|---------------------|----------------------|---------------|--------------------------------|--------------|-------------------|
| <input type="checkbox"/> | webserver1       | i-0c30c9909807edd2d | <span>Running</span> | t2.micro      | <span>2/2 checks passed</span> | No alarms    | ap-south-1a       |
|                          | webserver1-sc... | i-0a038cf7d341fe7d7 | <span>Running</span> | t2.micro      | <span>2/2 checks passed</span> | No alarms    | ap-south-1a       |

## 5. Added a record in route table of my domain for the web server load balancer.

Route 53 > Hosted zones > sumitmishra.info > Create record

### Quick create record Info

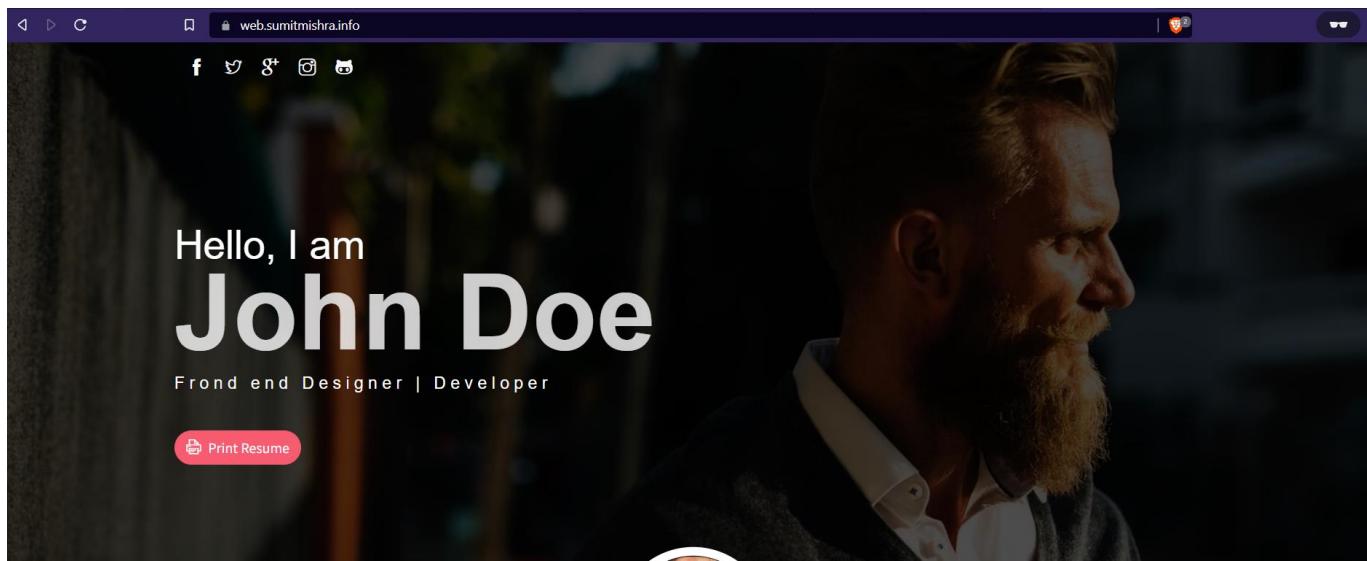
[Switch to wizard](#)

**Record 1**

|   |                                      |  |
|---|--------------------------------------|--|
| Record name <small>Info</small>   | .sumitmishra.info                    | Record type <small>Info</small>                              |
| <input type="text" value="web"/>  | .sumitmishra.info                    | A – Routes traffic to an IPv4 address and some AWS resources |
| Keep blank to create a record for the root domain.  |                                      |  |
| <input checked="" type="radio"/> Alias  |                                      |  |
| Route traffic to <small>Info</small>  |                                      |  |
| Alias to Application and Classic Load Balancer  |                                      |  |
| Asia Pacific (Mumbai) [ap-south-1]  |                                      |  |
| <input type="text" value="dualstack.webserver1-lb-890955047.ap-south-1.elb.amazonaws.com"/> |                                      |  |
| Routing policy <small>Info</small>  | Evaluate target health               |  |
| <input type="text" value="Simple routing"/>   | <input checked="" type="radio"/> Yes |  |
| <a href="#">Add another record</a>  |                                      |  |

[Cancel](#) [Create records](#)

## 6. Then I created an SSL certificate for the webserver using AWS certificate manager in the same way as I did for “personal-vpc-lb” (mentioned earlier).



As we can see here the SSL certificate is enabled for <https://web.sumitmishra.info> .

7. Created two Cloud watch alarms using the auto scaling group I just created named as “webserver1-asg” one for scaling out and another for scaling in.

CloudWatch > Alarms > Create alarm

Step 1  
**Specify metric and conditions**

Step 2  
Configure actions

Step 3  
Add name and description

Step 4  
Preview and create

### Specify metric and conditions

**Metric** Edit

**Graph**  
This alarm will trigger when the blue line goes above the red line for 1 datapoints within 1 minute.

Percent

CPUUtilization

Namespace  
AWS/EC2

Metric name  
CPUUtilization

AutoScalingGroupName  
webserver1-asg

Statistic  
Average

Period  
1 minute

## For scale-out

### Conditions

Threshold type

Static  
Use a value as a threshold

Anomaly detection  
Use a band as a threshold

Whenever CPUUtilization is...

Define the alarm condition.

Greater  
> threshold

Greater/Equal  
 $\geq$  threshold

Lower/Equal  
 $\leq$  threshold

Lower  
< threshold

than...

Define the threshold value.

70

Must be a number

## For scale in

### Conditions

Threshold type

Static  
Use a value as a threshold

Anomaly detection  
Use a band as a threshold

Whenever CPUUtilization is...

Define the alarm condition.

Greater  
> threshold

Greater/Equal  
 $\geq$  threshold

Lower/Equal  
 $\leq$  threshold

Lower  
< threshold

than...

Define the threshold value.

30

Must be a number

► Additional configuration

[Cancel](#) [Next](#)

Alarms created :

| Alarms (2)                  |           | <input type="checkbox"/> Hide Auto Scaling alarms | <input type="button" value="Clear selection"/> | <input type="button" value="C"/>                     | <input type="button" value="Create composite alarm"/> | <input type="button" value="Actions"/> | <input type="button" value="Create alarm"/> |
|-----------------------------|-----------|---|--|--|---|--|---|
| <input type="text"/> Search |           | <input type="button" value="Any state"/>          | <input type="button" value="Any type"/>        | <input type="button" value="Any actions ..."/>       | < 1 >   |  |   |
| <input type="checkbox"/>    | Name      | State   | Last state update                              | Conditions   | Actions   |  |   |
| <input type="checkbox"/>    | Scale-In  | In alarm  | 2022-08-01 11:48:28                            | CPUUtilization < 30 for 1 datapoints within 1 minute | No actions  |  |   |
| <input type="checkbox"/>    | Scale-Out | OK  | 2022-08-01 11:41:47                            | CPUUtilization > 70 for 1 datapoints within 1 minute | No actions  |  |   |

8. Then I created two dynamic scaling policies for scaling in and scaling out operation, and attached the alarms to respective scaling policies.

For scale-in:

EC2 > Auto Scaling groups > webserver1-asg

## Create dynamic scaling policy

Policy type

Simple scaling

Scaling policy name

scale-in

CloudWatch alarm

Choose an alarm that can scale capacity whenever:

Scale-In

[Create a CloudWatch alarm](#)

breaches the alarm threshold: CPUUtilization < 30 for 1 consecutive periods of 60 seconds for the metric dimensions:

AutoScalingGroupName = webserver1-asg

Take the action

Remove ▾ 1 capacity units ▾

And then wait

20 seconds before allowing another scaling activity

For scale-out :

## Edit dynamic scaling policy

Policy type

Simple scaling

Scaling policy name

scale-out

CloudWatch alarm

Choose an alarm that can scale capacity whenever:

Scale-Out

Create a CloudWatch alarm [\[edit\]](#)

breaches the alarm threshold: CPUUtilization > 70 for 1 consecutive periods of 60 seconds for the metric dimensions:

AutoScalingGroupName = webserver1-asg

Take the action

Add 1 capacity units

And then wait

20 seconds before allowing another scaling activity

[Cancel](#) [Update](#)

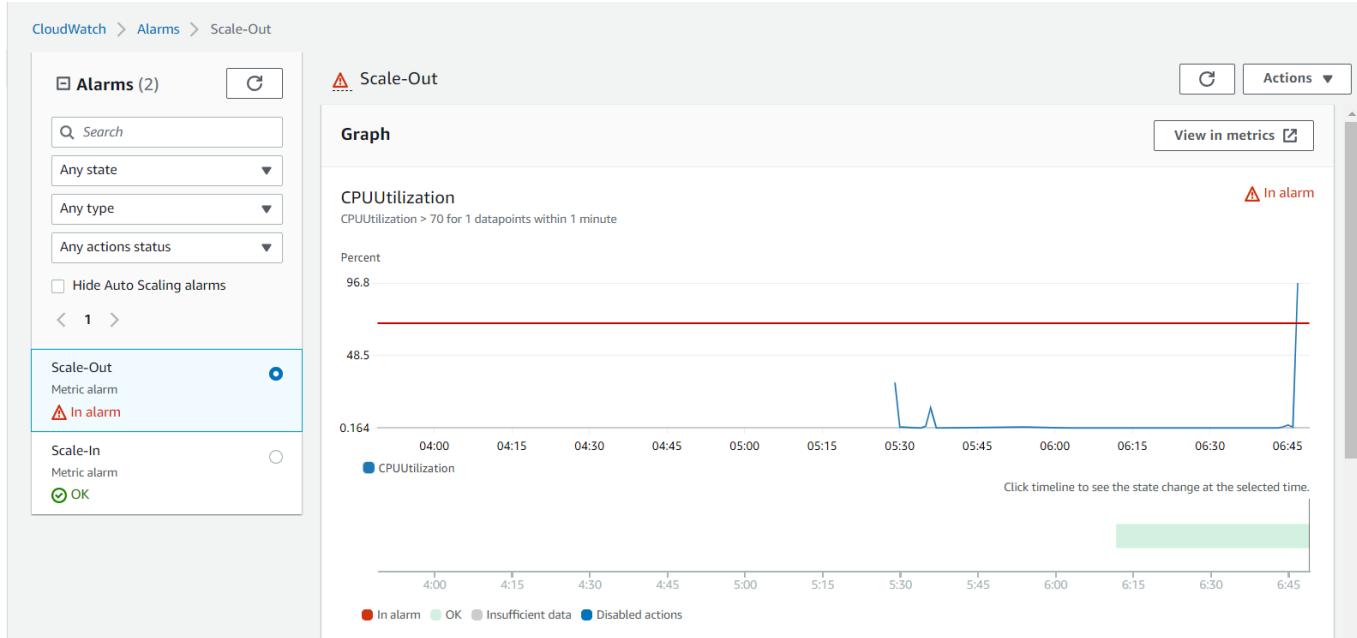
9. Then I increased the CPU utilization to 99% by typing the command “yes /dev/null &” then I typed “top” to see the utilization table.

```
[root@ip-7-0-1-148 ~]# yes > /dev/null &
[1] 3878
[root@ip-7-0-1-148 ~]# top
top - 06:23:59 up 1:57, 1 user,  load average: 0.66, 0.24, 0.09
Tasks: 108 total, 2 running, 64 sleeping, 0 stopped, 0 zombie
%Cpu(s): 98.7 us, 1.3 sy, 0.0 ni, 0.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 988688 total, 616948 free, 112272 used, 259468 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 729984 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
 3878 root      20   0 114644  760  696 R 99.7  0.1  0:33.75 yes
2945 apache    20   0 555948 8348 5260 S  0.3  0.8  0:00.94 httpd
2948 apache    20   0 326376 7100 4136 S  0.3  0.7  0:00.81 httpd
  1 root      20   0 41588 5264 3816 S  0.0  0.5  0:02.43 systemd
  2 root      20   0 0 0 0 S  0.0  0.0  0:00.00 kthreadd
  3 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 rcu_gp
  4 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 rcu_par_gp
  6 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 kworker/0:0H-ev
  8 root      0 -20 0 0 0 I  0.0  0.0  0:00.11 kworker/0:1H-ev
  9 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 mm_percpu_wq
 10 root     20   0 0 0 0 S  0.0  0.0  0:00.00 rcu_tasks_rude_
 11 root     20   0 0 0 0 S  0.0  0.0  0:00.00 rcu_tasks_trace
 12 root     20   0 0 0 0 S  0.0  0.0  0:00.03 ksoftirqd/0
 13 root     20   0 0 0 0 I  0.0  0.0  0:00.09 rcu_sched
 14 root      rt  0 0 0 0 S  0.0  0.0  0:00.03 migration/0
 16 root     20   0 0 0 0 S  0.0  0.0  0:00.00 cpuhp/0
 18 root     20   0 0 0 0 S  0.0  0.0  0:00.00 kdevtmpfs
 19 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 netns
 22 root     20   0 0 0 0 S  0.0  0.0  0:00.01 kaudit
264 root     20   0 0 0 0 S  0.0  0.0  0:00.00 khungtaskd
265 root     20   0 0 0 0 S  0.0  0.0  0:00.00 oom_reaper
266 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 writeback
268 root     20   0 0 0 0 S  0.0  0.0  0:00.16 kcompactd0
269 root      25   5 0 0 0 S  0.0  0.0  0:00.00 ksmd
270 root      39   19 0 0 0 S  0.0  0.0  0:00.00 khugepaged
325 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 kintegrityd
327 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 kblockd
328 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 blkcg_punt_bio
680 root     20   0 0 0 0 S  0.0  0.0  0:00.00 xen-balloon
686 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 tpm_dev_wq
692 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 md
695 root      0 -20 0 0 0 I  0.0  0.0  0:00.00 edac-poller
700 root     -51   0 0 0 0 S  0.0  0.0  0:00.00 watchdogd
849 root     20   0 0 0 0 S  0.0  0.0  0:00.00 kswapd0
```

ssh://ec2-user@3.111.96.10:22

Here, we can see that CPU utilization percentage is 99.7% and scale-out alarm is triggered.



10. In the backend, servers started to deploy automatically to manage the traffic of CPU.

| Instances (1/6) <a href="#">Info</a> |                  |                     |                      |                 |                       |                                |                   |                 |             | <a href="#">C</a> | <a href="#">Connect</a> | <a href="#">Instance state ▾</a> | <a href="#">Actions ▾</a> | <a href="#">Launch instances</a> | <a href="#">▼</a> |
|--------------------------------------|------------------|---------------------|----------------------|-----------------|-----------------------|--------------------------------|-------------------|-----------------|-------------|-------------------|-------------------------|----------------------------------|---------------------------|----------------------------------|-------------------|
|                                      | Name             | Instance ID         | Instance state       | Instance type   | Status check          | Alarm status                   | Availability Zone | Public IPv4 DNS | Public IP   |                   |                         |                                  |                           |                                  |                   |
| <input type="checkbox"/>             | webserver1       | i-0c30c9909807edd2d | <span>Running</span> | <span>QQ</span> | <span>t2.micro</span> | <span>2/2 checks passed</span> | No alarms         | <span>+</span>  | ap-south-1a | -                 |                         |                                  |                           | 3.111.96                         |                   |
| <input type="checkbox"/>             | webserver3       | i-0e0945972773e248c | <span>Stopped</span> | <span>QQ</span> | <span>t2.micro</span> | -                              | No alarms         | <span>+</span>  | ap-south-1a | -                 |                         |                                  |                           | 43.205.4                         |                   |
| <input type="checkbox"/>             | webserver1-sc... | i-0a038cf7d341fe7d7 | <span>Running</span> | <span>QQ</span> | <span>t2.micro</span> | <span>2/2 checks passed</span> | No alarms         | <span>+</span>  | ap-south-1a | -                 |                         |                                  |                           | 13.127.1                         |                   |
| <input type="checkbox"/>             | webserver2       | i-0bbe0940452be4101 | <span>Stopped</span> | <span>QQ</span> | <span>t2.micro</span> | -                              | No alarms         | <span>+</span>  | ap-south-1b | -                 |                         |                                  |                           | 3.109.30                         |                   |
| <input type="checkbox"/>             | webserver4       | i-02640c5161f8a7117 | <span>Stopped</span> | <span>QQ</span> | <span>t2.micro</span> | -                              | No alarms         | <span>+</span>  | ap-south-1b | -                 |                         |                                  |                           | 43.205.6                         |                   |
| <input checked="" type="checkbox"/>  | webserver1-sc... | i-0a461f60bcdb96504 | <span>Running</span> | <span>QQ</span> | <span>t2.micro</span> | <span>Initializing</span>      | No alarms         | <span>+</span>  | ap-south-1b | -                 |                         |                                  |                           | 13.234.6                         |                   |

11. After that I decreased the CPU utilization of the server using the command “killall -p yes”. Then used “top” command to get the metrics.

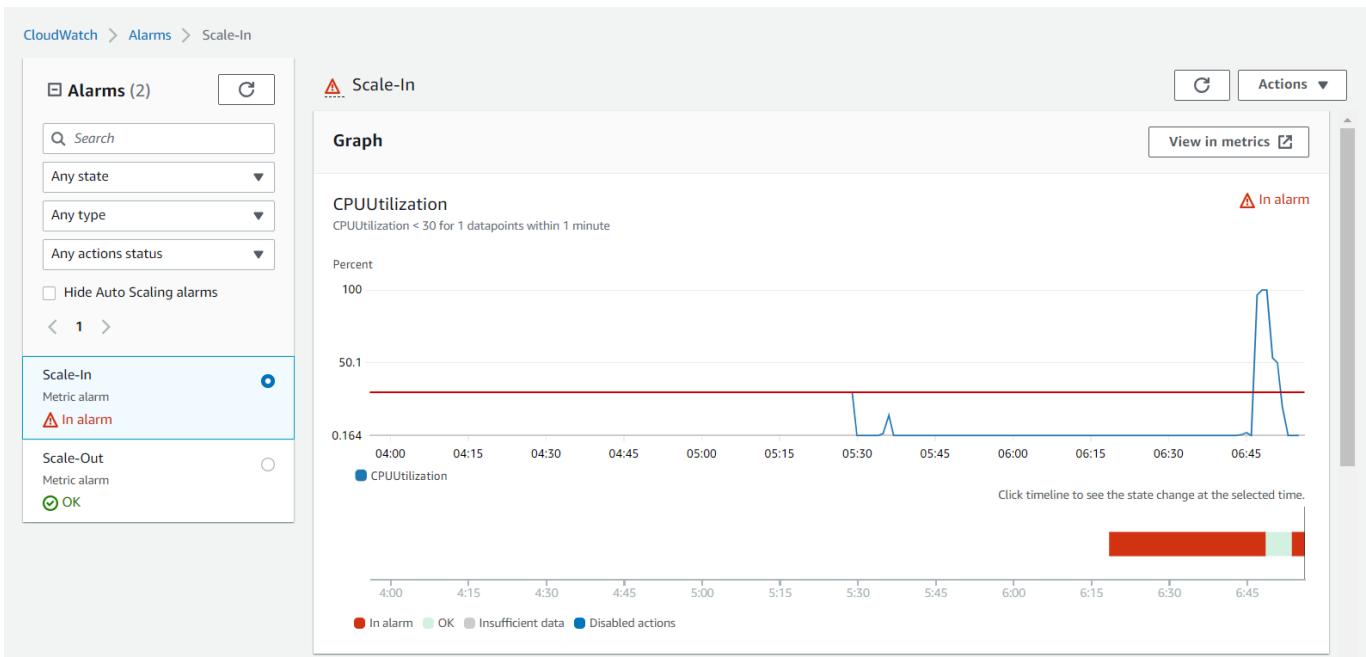
We can see that CPU utilization has dropped close to 0%.

```
[root@ip-7-0-1-206 ~]# killall -p yes
[root@ip-7-0-1-206 ~]# top
```

top - 06:53:09 up 1:24, 1 user, load average: 0.47, 0.57, 0.29
Tasks: 106 total, 1 running, 64 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
KiB Mem : 988688 total, 613868 free, 112408 used, 262412 buff/cache
KiB Swap: 0 total, 0 free, 0 used. 729572 avail Mem

| PID | USER | PR | NI  | VIRT   | RES  | SHR  | S | %CPU | %MEM | TIME+   | COMMAND         |
|-----|------|----|-----|--------|------|------|---|------|------|---------|-----------------|
| 1   | root | 20 | 0   | 123504 | 5396 | 3892 | S | 0.0  | 0.5  | 0:02.11 | systemd         |
| 2   | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | kthreadd        |
| 3   | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | rcu_gp          |
| 4   | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | rcu_par_gp      |
| 6   | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | kworker/0:0H-ev |
| 8   | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.07 | kworker/0:1H-ev |
| 9   | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | mm_percpu_wq    |
| 10  | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | rcu_tasks_rude_ |
| 11  | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | rcu_tasks_trace |
| 12  | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.03 | ksoftirqd/0     |
| 13  | root | 20 | 0   | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.11 | rcu_sched       |
| 14  | root | rt | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.02 | migration/0     |
| 16  | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | cpuhp/0         |
| 18  | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | kdevtmpfs       |
| 19  | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | netns           |
| 20  | root | 20 | 0   | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.02 | kworker/u30:1-e |
| 22  | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.01 | kauditd         |
| 264 | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | khungtaskd      |
| 265 | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | oom_reaper      |
| 266 | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | writeback       |
| 268 | root | 20 | 0   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.10 | kcompactd0      |
| 269 | root | 25 | 5   | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | ksmd            |
| 270 | root | 39 | 19  | 0      | 0    | 0    | S | 0.0  | 0.0  | 0:00.00 | khugepaged      |
| 325 | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | kiintegrityd    |
| 327 | root | 0  | -20 | 0      | 0    | 0    | I | 0.0  | 0.0  | 0:00.00 | kblockd         |

We can see that the scale in alarm has been triggered in the alarm section.



## 12. In the backend, the allocated servers started to terminate.

| Instances (1/6) <a href="#">Info</a> |                  |                             |                                |                               |                                |                              |                                   |                                 | <a href="#">C</a> | <a href="#">Connect</a> | <a href="#">Instance state ▾</a> | <a href="#">Actions ▾</a> | <a href="#">Launch instances</a> | <a href="#">▼</a> |
|--------------------------------------|------------------|-----------------------------|--------------------------------|-------------------------------|--------------------------------|------------------------------|-----------------------------------|---------------------------------|-------------------|-------------------------|----------------------------------|---------------------------|----------------------------------|-------------------|
| <a href="#">Name</a>                 |                  | <a href="#">Instance ID</a> | <a href="#">Instance state</a> | <a href="#">Instance type</a> | <a href="#">Status check</a>   | <a href="#">Alarm status</a> | <a href="#">Availability Zone</a> | <a href="#">Public IPv4 DNS</a> |                   | <a href="#">◀</a>       | <a href="#">1</a>                | <a href="#">▶</a>         | <a href="#">⊗</a>                |                   |
| <input type="checkbox"/>             | webserver1       | i-0c30c9909807edd2d         | <span>Running</span>           | t2.micro                      | <span>2/2 checks passed</span> | No alarms                    | ap-south-1a                       | -                               | 3.111.9           |                         |                                  |                           |                                  |                   |
| <input type="checkbox"/>             | webserver3       | i-0e0945972773e248c         | <span>Stopped</span>           | t2.micro                      | -                              | No alarms                    | ap-south-1a                       | -                               | 43.205.4          |                         |                                  |                           |                                  |                   |
| <input checked="" type="checkbox"/>  | webserver1-sc... | i-0a038cf7d341fe7d7         | <span>Shutting-down</span>     | t2.micro                      | -                              | No alarms                    | ap-south-1a                       | -                               | 13.127.1          |                         |                                  |                           |                                  |                   |
| <input type="checkbox"/>             | webserver2       | i-0bbe0940452be4101         | <span>Stopped</span>           | t2.micro                      | -                              | No alarms                    | ap-south-1b                       | -                               | 3.109.30          |                         |                                  |                           |                                  |                   |
| <input type="checkbox"/>             | webserver4       | i-02640c5161f8a7117         | <span>Stopped</span>           | t2.micro                      | -                              | No alarms                    | ap-south-1b                       | -                               | 43.205.6          |                         |                                  |                           |                                  |                   |
| <input type="checkbox"/>             | webserver1-sc... | i-0a461f60bcd96504          | <span>Running</span>           | t2.micro                      | <span>2/2 checks passed</span> | No alarms                    | ap-south-1b                       | -                               | 13.234.6          |                         |                                  |                           |                                  |                   |



8/5/2022

# Project-02

**Migrating an On-premises server to AWS.**

(On-premises server is taken to be a server in a  
different AWS region for this project.)



Sumit Mishra  
SIC: 190310286

# 1. Creating an On-premises server in a different region.

## a. Creating an EC2 instance inside a custom VPC with subnets, internet gateway enabled.

The screenshot shows the AWS EC2 Management Console interface. At the top, the region is set to "Singapore". In the main pane, there is one instance listed: "Instances (1/1) Info". The instance details are as follows:

| Name       | Instance ID        | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IPv4 DNS | Public IPv4 ... |
|------------|--------------------|----------------|---------------|--------------|--------------|-------------------|-----------------|-----------------|
| web-server | i-0d23f19a0fd22ed0 | Running        | t2.micro      | -            | No alarms    | ap-southeast-1b   | 54.169.117.71   |                 |

Below the instance list, a detailed view for "Instance: i-0d23f19a0fd22ed0 (web-server)" is shown. The "Details" tab is selected. Key information includes:

- Instance ID: i-0d23f19a0fd22ed0 (web-server)
- Public IPv4 address: 54.169.117.71
- Private IP4 address: 10.0.2.180
- Instance state: Running
- Private IP DNS name (IPv4 only): ip-10-0-2-180.ap-southeast-1.compute.internal
- Instance type: t2.micro

## b. Connecting to the EC2 instance and downloading/installing Apache server in it.

```
Xshell 7 (Build 0111)
Copyright (c) 2020 NetSarang Computer, Inc. All rights reserved.
Type 'help' to learn how to use Xshell prompt.
[IC:\->]$ ssh -i "singapore-key.pem" ec2-user@54.169.117.71

Connecting to 54.169.117.71:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+]'.

WARNING! The remote SSH server rejected X11 forwarding request.

  _\   _\_) /  Amazon Linux 2 AMI
  \_\_|_/_|_|
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-10-0-2-180 ~]$ sudo su
[root@ip-10-0-2-180 ec2-user]# cd
[root@ip-10-0-2-180 ~]# yum install httpd -y
Loaded plugins: extras_suggestions, langpacks, priorities, update-motd
amzn2-core
Resolving Dependencies
--> Running transaction check
--> Package httpd.x86_64 0:2.4.54-1.amzn2 will be installed
--> Processing Dependency: httpd-tools = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem = 2.4.54-1.amzn2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: system-logos-htpd for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: mod_http2 for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: httpd-filesystem for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: /etc/mime.types for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libaprutil-1.so.0()(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Processing Dependency: libapr-1.so.0()(64bit) for package: httpd-2.4.54-1.amzn2.x86_64
--> Running transaction check
--> Package apr.x86_64 0:1.7.0-9.amzn2 will be installed
--> Package apr-util.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Processing Dependency: apr-util-bdb(x86-64) = 1.6.1-5.amzn2.0.2 for package: apr-util-1.6.1-5.amzn2.0.2.x86_64
--> Package generic-logos-htpd.noarch 0:2.4.54-1.amzn2 will be installed
--> Package httpd-filesystem.noarch 0:2.4.54-1.amzn2 will be installed
--> Package httpd-tools.x86_64 0:2.4.54-1.amzn2 will be installed
--> Package mailcap.noarch 0:2.1.41-2.amzn2 will be installed
--> Package mod_http2.x86_64 0:1.19.19-1.amzn2.0.1 will be installed
--> Running transaction check
--> Package apr-util-bdb.x86_64 0:1.6.1-5.amzn2.0.2 will be installed
--> Finished Dependency Resolution
```

c. Putting some sample website to display.

```
[root@ip-10-0-2-180 ~]# wget https://www.free-css.com/assets/files/free-css-templates/download/page281/limelight.zip
--2022-08-05 06:46:41--  https://www.free-css.com/assets/files/free-css-templates/download/page281/limelight.zip
Resolving www.free-css.com (www.free-css.com)... 217.160.0.242, 2001:8d8:100f:f000::28f
Connecting to www.free-css.com (www.free-css.com)|217.160.0.242|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 6646619 (6.3M) [application/zip]
Saving to: 'limelight.zip'

100%[=====] 6,646,619  3.23MB/s  in 2.0s

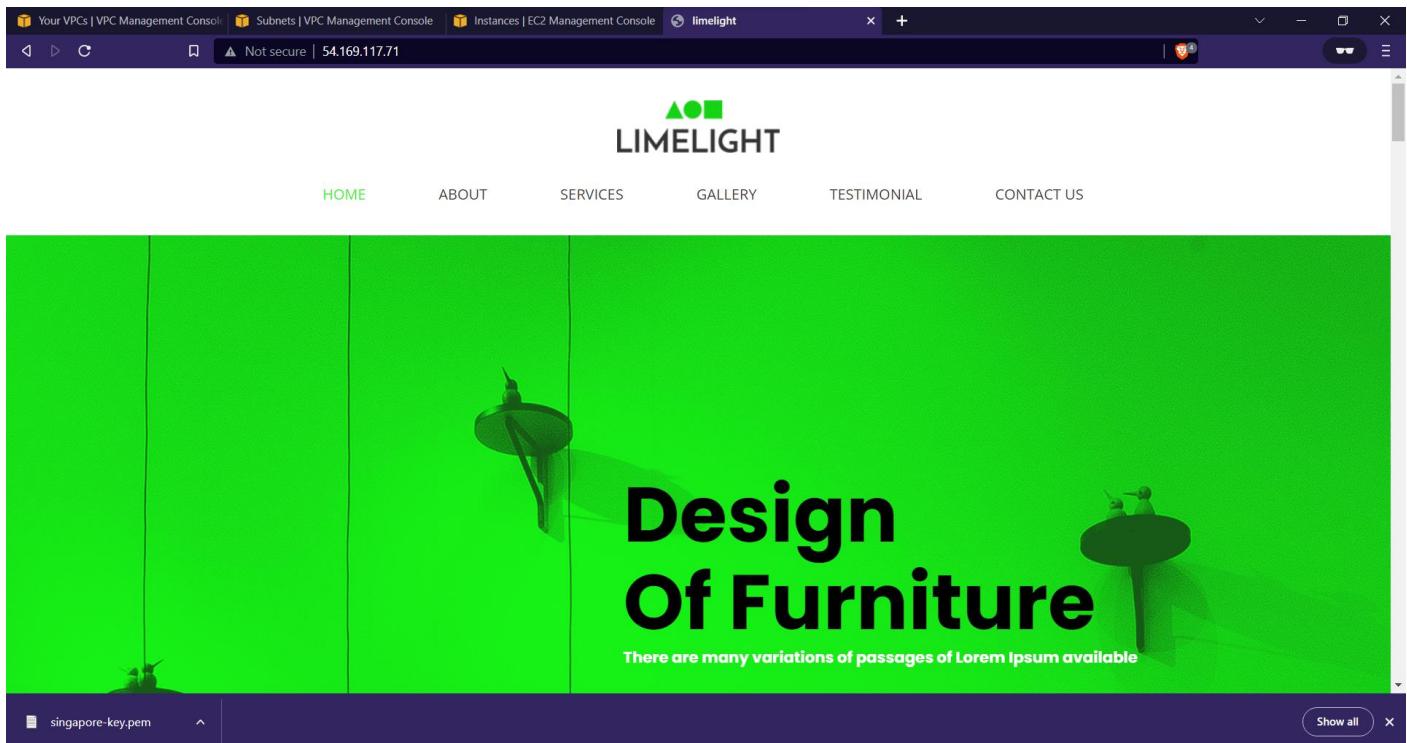
2022-08-05 06:46:45 (3.23 MB/s) - 'limelight.zip' saved [6646619/6646619]

[root@ip-10-0-2-180 ~]# unzip limelight.zip
Archive: limelight.zip
  creating: limelight-html/
  inflating: limelight-html/about.html
  inflating: limelight-html/contact.html
  creating: limelight-html/css/
  inflating: limelight-html/css/.DS_Store
```

### limelight-html limelight.zip

```
[root@ip-10-0-2-180 ~]# ls limelight-html/
about.html  css  gallery.html  images  js          testimonial.html
contact.html  fonts  icon        index.html  service.html
[root@ip-10-0-2-180 ~]# cp -r limelight-html/* /var/www/html
[root@ip-10-0-2-180 ~]#
[root@ip-10-0-2-180 ~]# 
```

```
[root@ip-10-0-2-180 ~]# systemctl start httpd.service
[root@ip-10-0-2-180 ~]# systemctl enable httpd.service
Created symlink from /etc/systemd/system/multi-user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
[root@ip-10-0-2-180 ~]# 
```



2. Migrating the server On-premises to AWS (i.e., from Singapore region to Mumbai region).

a. Set up application migration service and create template.

No servers

Add your source servers to this console by installing the AWS Replication Agent on them or by installing the AWS vCenter client on your vCenter in order to add source servers without installing an agent on each guest server

Add servers

b. Adding server configuration for migration.  
(1) Creating an IAM role for server migration.

| User name           | Groups | Last activity | MFA  | Password age | Active key age |
|---------------------|--------|---------------|------|--------------|----------------|
| mgn-agent-installer | None   | Never         | None | None         | Now            |

Name – Sumit Mishra  
SIC - 190310286

(2) Provide the details of the IAM user.

1. Select your operating system

Linux

Windows

Legacy OS: Windows Server 2003 or Windows Server 2008

2. Select your replication preferences [Info](#)

Replicate all disks

3. IAM access key ID [Info](#)

AKIATYFGC35IEIFGI4IC

[Create IAM user](#)

IAM secret access key

This form does not send the secret – it only adds it to the installation command you can copy

.....

[Show](#)

(3) Download the installer using the copied command.

```
[root@ip-10-0-2-180 ~]# wget -O ./aws-replication-installer-init.py https://aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com/latest/linux/aws-replication-installer-init.py
--2022-08-05 07:06:00-- https://aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com/latest/linux/aws-replication-installer-init.py
Resolving aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com (aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com)... 52.219.64.119
Connecting to aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com (aws-application-migration-service-ap-south-1.s3.ap-south-1.amazonaws.com)|52.219.64.119|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 14387 (14K) [binary/octet-stream]
Saving to: './aws-replication-installer-init.py'

100%[=====] 14,387      --K/s   in 0s

2022-08-05 07:06:00 (136 MB/s) - './aws-replication-installer-init.py' saved [14387/14387]
```

(4) Download the replication agent using the copied command.

```
[root@ip-10-0-2-180 ~]#
[root@ip-10-0-2-180 ~]# sudo python3 aws-replication-installer-init.py --region ap-south-1 --aws-access-key-id AKIATYFGC35IEIFGI4IC --aws-secret-access-key 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 --no-prompt
The installation of the AWS Replication Agent has started.
Identifying volumes for replication.
Identified volume for replication: /dev/xvda of size 8 GiB
All volumes for replication were successfully identified.
Downloading the AWS Replication Agent onto the source server... Finished.
Installing the AWS Replication Agent onto the source server... Finished.
Syncing the source server with the Application Migration Service Console... Finished.
The following is the source server ID: s-e38c7e2c9bff08a89.
You now have 1 active source server out of a total quota of 20.
Learn more about increasing source servers limit at https://docs.aws.amazon.com/mgn/latest/ug/MGN-service-limits.html
The AWS Replication Agent was successfully installed.
[root@ip-10-0-2-180 ~]#
```

c. After adding source server to the migration service, wait for the replication to complete.

| Source servers (1)                            |          |  |                       |                               |                 |                                   |
|---|----------|--|-----------------------|-------------------------------|-----------------|-----------------------------------|
| Actions ▾                                     |          | Replication ▾                              |                       | Test and cutover ▾            |                 |                                   |
| Active source servers                         |          | Filter source servers by property or value |                       |                               | < 1 > ⚙         |                                   |
| Source server name ▾                          | Alerts ▾ | Replication type ▾                         | Migration lifecycle ▾ | Data replication status       | Last snapshot ▾ | Next step ▾                       |
| ip-10-0-2-180.ap-southeast-1.compute.internal | -        | Agent based                                | Not ready             | Initial sync 7%   19 min left | -               | Wait for initial sync to complete |

This account is currently replicating 1 server out of a quota of 20 concurrent replicating servers. [Learn more](#)

d. After the replication completed, I launched the test instance as specified in the next step.

| Source servers (1)                            |          |  |                       |                         |                 |             |
|---|----------|--|-----------------------|-------------------------|-----------------|-------------|
| Actions ▾                                     |          | Replication ▾                              |                       | Test and cutover ▾      |                 |             |
| Active source servers                         |          | Filter source servers by property or value |                       |                         | < 1 > ⚙         |             |
| Source server name ▾                          | Alerts ▾ | Replication type ▾                         | Migration lifecycle ▾ | Data replication status | Last snapshot ▾ | Next step ▾ |
| ip-10-0-2-180.ap-southeast-1.compute.internal | -        | Agent based                                | Ready for testing     | Healthy                 | -               |             |

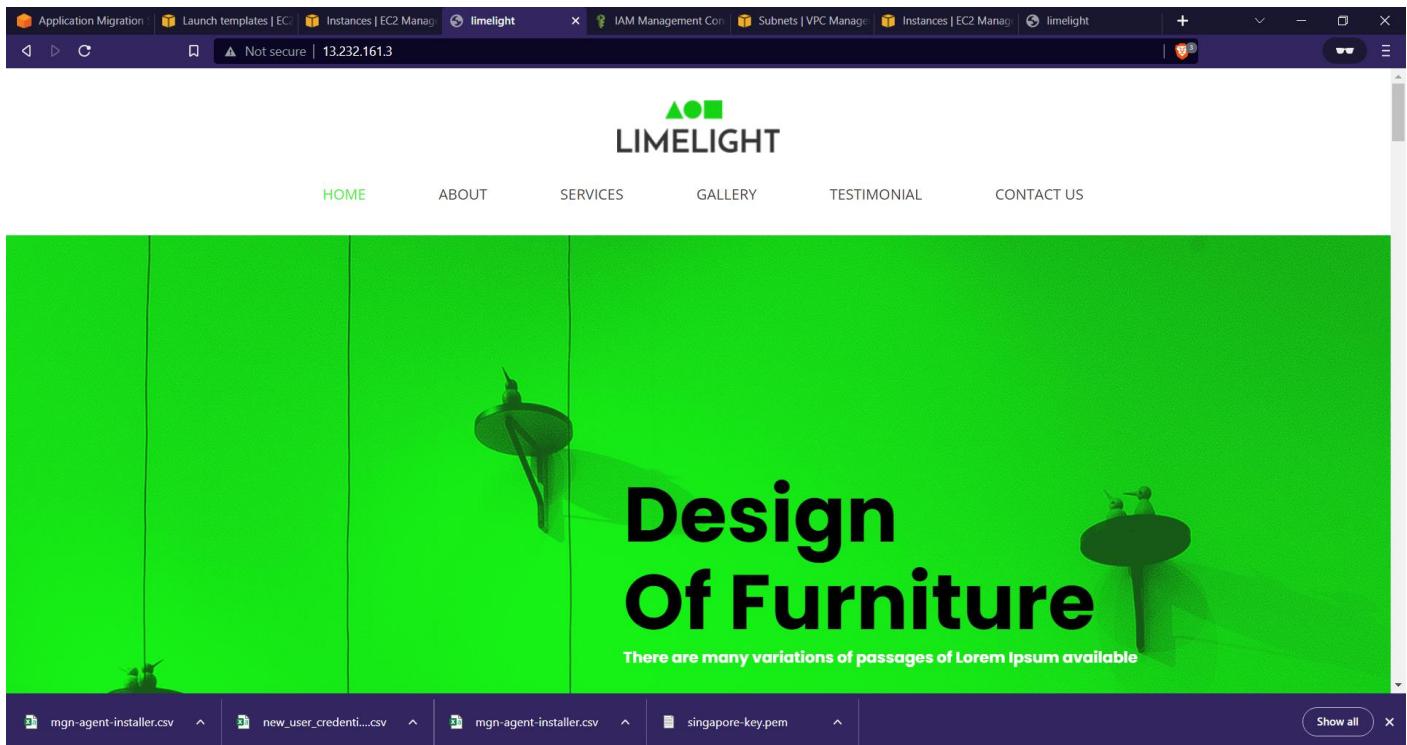
This account is currently replicating 1 server out of a quota of 20 concurrent replicating servers. [Learn more](#)

- Testing**
- Launch test instances
- Mark as "Ready for cutover"
- Revert to "Ready for testing"
- Cutover**
- Launch cutover instances
- Finalize cutover
- Revert to "Ready for cutover"
- Other**

| Instances (3) <a href="#">Info</a>                   |                     |                  |               |                   |              |                    |
|--|---------------------|------------------|---------------|-------------------|--------------|--------------------|
| Actions ▾  |                     | Instance state ▾ |               | Actions ▾         |              | Launch instances ▾ |
| <a href="#">Search</a>                               |                     |                  |               |                   |              |                    |
| Name   | Instance ID         | Instance state   | Instance type | Status check      | Alarm status |                    |
| AWS Application Migration Service Conversion Server  | i-0b399509856b72981 | Terminated       | m4.large      | -                 | No alarms    | +                  |
| ip-10-0-1-111.ap-southeast-1.compute.internal        | i-0a97812284b7c2cd9 | Running          | t2.micro      | Initializing      | No alarms    | +                  |
| AWS Application Migration Service Replication Server | i-03b5e595b88f97865 | Running          | t2.micro      | 2/2 checks passed | No alarms    | +                  |

The launched test server reflected on the EC2 console.

e. Tested using the public IP of ip-10-0-1-111.ap-southeast-1.compute.internal instance to check if conversion was properly completed or not.



As the website was able to host on the public IP, I concluded that the conversion had been done properly.

- After testing was successful, I marked the server as “ready for cut-over” as was mentioned in the next step column.

| Source servers (2)                            |  |                  |                     |                         | Actions ▾   | Replication ▾ | Test and cutover ▾ |
|---|--|------------------|---------------------|-------------------------|---|---------------|--------------------|
| Active source servers                         |  |                  |                     |                         | <input type="text"/> Filter source servers by property or value   |               |                    |
| Source server name                            | Alerts ▾                                     | Replication type | Migration lifecycle | Data replication status |   |               |                    |
| ip-10-0-1-111.ap-southeast-1.compute.internal | <input checked="" type="checkbox"/> Launched | Agent based      | Test in progress    | Healthy                 | <a href="#">Testing</a><br><a href="#">Launch test instances</a><br><b>Mark as "Ready for cutover"</b><br><a href="#">Revert to "Ready for testing"</a><br><br><a href="#">Cutover</a><br><a href="#">Launch cutover instances</a><br><a href="#">Finalize cutover</a><br><a href="#">Revert to "Ready for cutover"</a><br><br><a href="#">Other</a><br><a href="#">Edit Launch Settings</a><br><a href="#">Edit post-launch settings</a><br><a href="#">Terminate launched instances</a> |               |                    |
| ip-10-0-2-180.ap-southeast-1.compute.internal | <input type="checkbox"/> Launched            | Agent based      | Disconnected        | Disconnected            |   |               |                    |

This account is currently replicating 2 servers out of a quota of 20 concurrent replicating servers. [Learn more](#)

- After I marked the server as “ready for cutover”, I then launched cutover instances as was mentioned in the next step.

Source servers (2)

Actions ▾ Replication ▾ Test and cutover ▾

Active source servers ▾ Filter source servers by property or value

| Source server name                            | Alerts                                       | Replication type | Migration lifecycle | Data replication status |
|---|--|------------------|---------------------|-------------------------|
| ip-10-0-1-111.ap-southeast-1.compute.internal | -  | Agent based      | Ready for cutover   | Healthy                 |
| ip-10-0-2-180.ap-southeast-1.compute.internal | <input checked="" type="checkbox"/> Launched | Agent based      | Disconnected        | Disconnected            |

This account is currently replicating 2 servers out of a quota of 20 concurrent replicating servers. [Learn more](#)

Testing  
Launch test instances  
Mark as "Ready for cutover"  
Revert to "Ready for testing"

Cutover  
Launch cutover instances  
Finalize cutover  
Revert to "Ready for cutover"

Other  
Edit Launch Settings  
Edit post-launch settings

I tested the cut-over server and it worked fine.

Instances (1/5) [Info](#)

Search

| Name   | Instance ID         | Instance state | Instance type | Status check      | Alarm status |
|--|---------------------|----------------|---------------|-------------------|--------------|
| AWS Application Migration Service Conversion Server  | i-0b399509856b72981 | Terminated     | m4.large      | -                 | No alarms +  |
| ip-10-0-1-111.ap-southeast-1.compute.internal        | i-0a97812284b7c2cd9 | Terminated     | t2.micro      | -                 | No alarms +  |
| AWS Application Migration Service Replication Server | i-03b5e595b88f97865 | Running        | t2.micro      | 2/2 checks passed | No alarms +  |
| AWS Application Migration Service Conversion Server  | i-0a72391567d94e445 | Terminated     | m4.large      | -                 | No alarms +  |
| ip-10-0-1-111.ap-southeast-1.compute.internal        | i-035d9da40930c9791 | Running        | t2.micro      | 2/2 checks passed | No alarms +  |

Instance: i-035d9da40930c9791 (ip-10-0-1-111.ap-southeast-1.compute.internal)

Details Security Networking Storage Status checks Monitoring Tags

Instance summary [Info](#)

Instance ID: i-035d9da40930c9791 (ip-10-0-1-111.ap-southeast-1.compute.internal)

Public IPv4 address copied

Private IPv4 addresses: 7.0.1.245

IPv6 address: -

Instance state: Running

Public IPv4 DNS: -

Application Migration | Launch templates | EC2 | Instances | EC2 Manager | limelight | IAM Management Console | Subnets | VPC Manager | Instances | EC2 Manager | limelight

Not secure | 13.232.2.131

LIMELIGHT

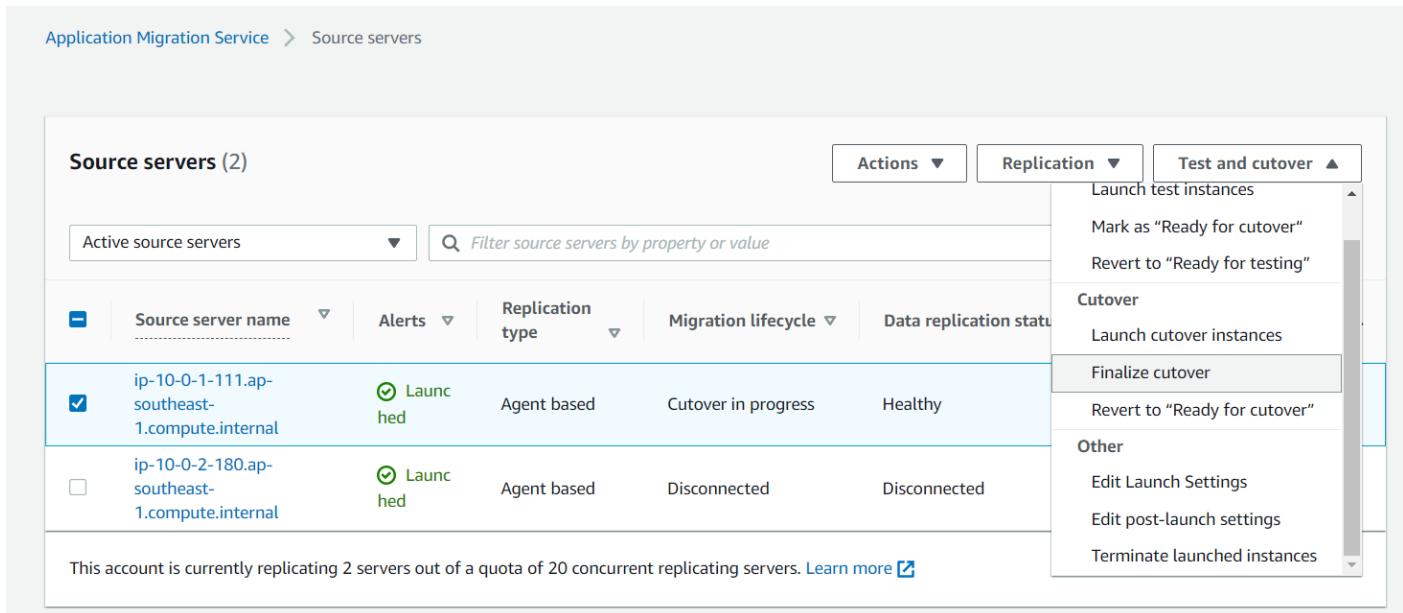
HOME ABOUT SERVICES GALLERY TESTIMONIAL CONTACT US

# Design Of Furniture

There are many variations of passages of Lorem Ipsum available.

READ MORE

h. After testing the cut-over instances, I finalized the cut-over as was mentioned in the next step.



Source servers (2)

Active source servers

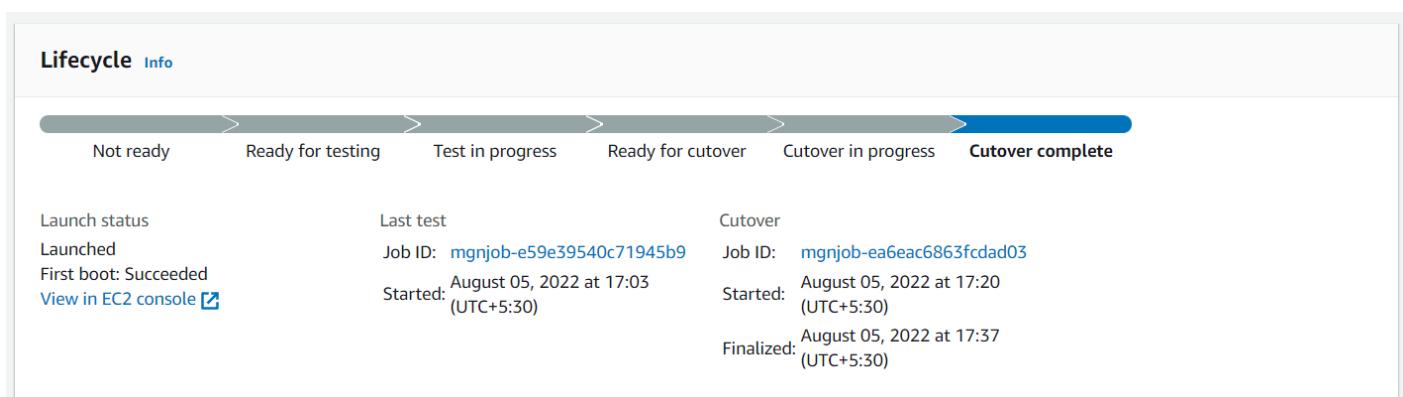
Filter source servers by property or value

| Source server name                            | Replication type | Migration lifecycle | Data replication status |
|---|------------------|---------------------|-------------------------|
| ip-10-0-1-111.ap-southeast-1.compute.internal | Launched         | Agent based         | Cutover in progress     |
| ip-10-0-2-180.ap-southeast-1.compute.internal | Launched         | Agent based         | Disconnected            |

This account is currently replicating 2 servers out of a quota of 20 concurrent replicating servers. [Learn more](#)

Actions ▾ Replication ▾ Test and cutover ▾

- Launch test instances
- Mark as "Ready for cutover"
- Revert to "Ready for testing"
- Cutover
  - Launch cutover instances
  - Finalize cutover
  - Revert to "Ready for cutover"
- Other
  - Edit Launch Settings
  - Edit post-launch settings
  - Terminate launched instances

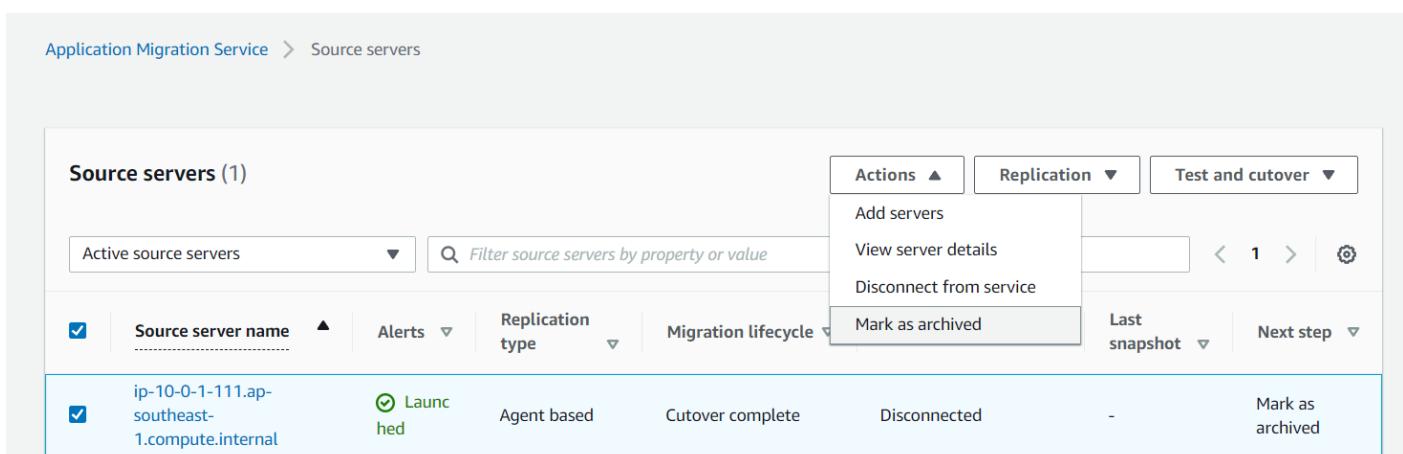


Lifecycle Info

Not ready > Ready for testing > Test in progress > Ready for cutover > Cutover in progress > **Cutover complete**

| Launch status                       | Last test  | Cutover   |
|-------------------------------------|--|---|
| Launched                            | Job ID: <a href="#">mgnjob-e59e39540c71945b9</a> | Job ID: <a href="#">mgnjob-ea6eac6863fcad03</a> |
| First boot: Succeeded               | Started: August 05, 2022 at 17:03 (UTC+5:30)     | Started: August 05, 2022 at 17:20 (UTC+5:30)    |
| <a href="#">View in EC2 console</a> |  | Finalized: August 05, 2022 at 17:37 (UTC+5:30)  |

i. After the cut-over was completed, I was marked the server as archived as was mentioned in the next step column.



Source servers (1)

Active source servers

Filter source servers by property or value

| Source server name                            | Replication type | Migration lifecycle | Data replication status | Next step    |
|---|------------------|---------------------|-------------------------|--------------|
| ip-10-0-1-111.ap-southeast-1.compute.internal | Launched         | Agent based         | Cutover complete        | Disconnected |

Actions ▾ Replication ▾ Test and cutover ▾

- Add servers
- View server details
- Disconnect from service
- Mark as archived

Last snapshot ▾ Next step ▾

Now, Mumbai region also has the exact copy of the web-server present in Singapore, so server migration was successful from on-premises to AWS.



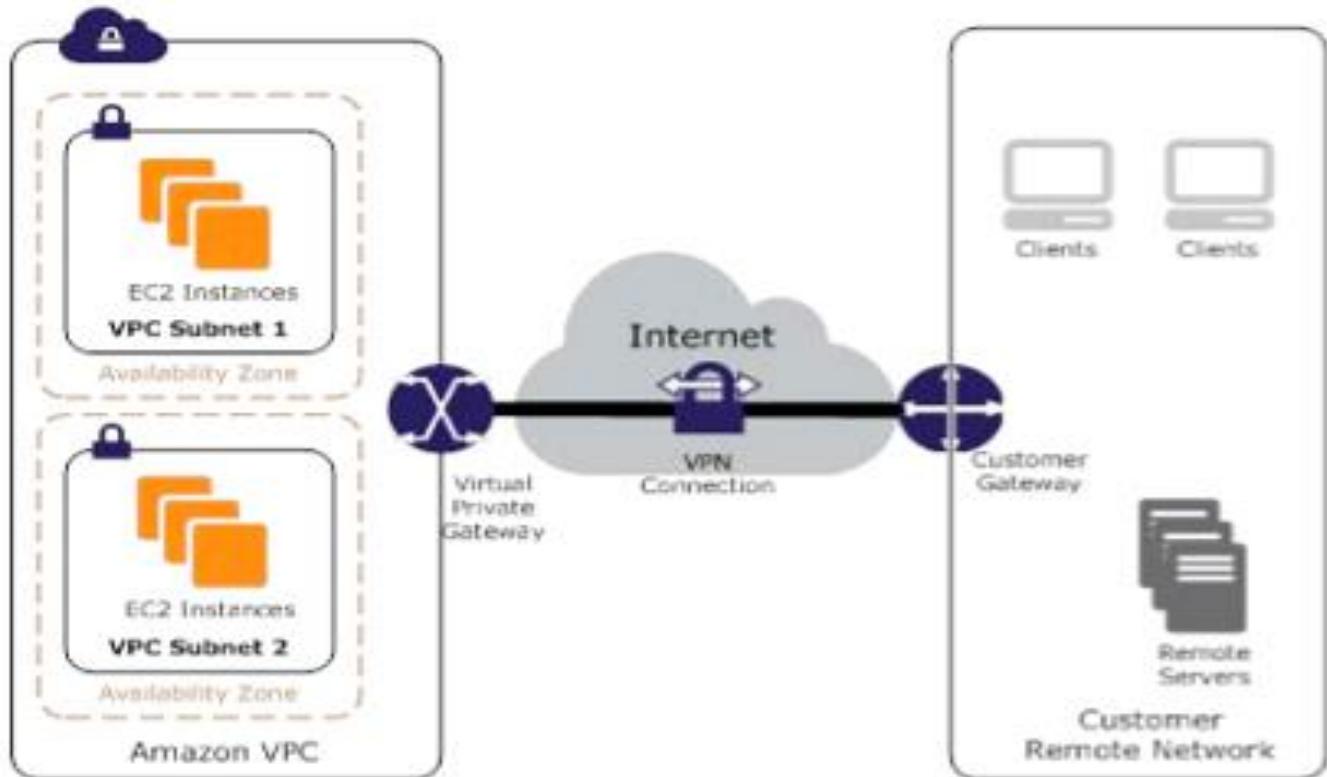
8/5/2022

# Project-03

1. Configuring Site to Site Connectivity on AWS.
2. Configuring Point to Site Connectivity on AWS.
3. Configuring Transit Gateway.

Sumit Mishra  
SIC: 190310286

## 1. Configuring Site to Site connectivity on AWS. (AWS site only)



**Figure 1: Hardware VPN**

1. Configured Virtual Private Gateway for a VPC which acts as site 1.

| Virtual private gateways (1) <a href="#">Info</a> |                            |          |         |     |            |  | <a href="#">Actions</a> | <a href="#">Create virtual private gateway</a> |
|---|----------------------------|----------|---------|-----|------------|--|-------------------------|--|
| Name  | Virtual private gateway ID | State    | Type    | VPC | Amazon ASN |  |                         |  |
| vpg-1   | vgw-030546fe44ba7bc7e      | Detached | ipsec.1 | -   | 64512      |  |                         |  |

2. Attached the vpg-1 to Mumbai-vpc.

| ⌚ You successfully attached vgw-030546fe44ba7bc7e / vpg-1 to vpc-083ef3fe083e0f251. | X                          |          |         |                                |
|---|----------------------------|----------|---------|--------------------------------|
| <a href="#">Virtual private gateways (1/1) <a href="#">Info</a></a>                 |                            |          |         |                                |
| <input type="text"/> Filter virtual private gateways                                |                            |          |         |                                |
| Name  | Virtual private gateway ID | State    | Type    | VPC                            |
| vpg-1   | vgw-030546fe44ba7bc7e      | Attached | ipsec.1 | vpc-083ef3fe083e0f251   mum... |

3. Created a Customer Gateway to attach to on-premises network acting as site 2.

✓ You successfully created cgw-0b5f8de3e17c14fed / cg-1.

X

| Customer gateways (1) <a href="#">Info</a>  |                       |           |         |  |   |
|---|-----------------------|-----------|---------|--|---|
| <input type="button" value="C"/> Actions ▾ <a href="#">Create customer gateway</a>            |                       |           |         |  |   |
| <input type="text"/> Filter customer gateways <span style="float: right;">&lt; 1 &gt; </span> |                       |           |         |  |   |
| <b>Customer gateway ID:</b> cgw-0b5f8de3e17c14fed <input type="button" value="X"/>            |                       |           |         | <input type="button" value="Clear filters"/> |   |
| Name ▾  | Customer gateway ID   | State     | BGP ASN | IP address                                   | ▼ |
| <input checked="" type="radio"/> cg-1   | cgw-0b5f8de3e17c14fed | Available | 65000   | 7.7.7.7                                      |   |

#### 4. Creating a Site to site connection using Site-to-Site VPN gateway.

| VPN connections (1/1) <a href="#">Info</a>  |                       |         |                           |  |   |
|---|-----------------------|---------|---------------------------|--|---|
| <input type="button" value="C"/> Actions ▾ <a href="#">Download configuration</a> <a href="#">Create VPN connection</a> |                       |         |                           |  |   |
| <input type="text"/> Filter VPN connections <span style="float: right;">&lt; 1 &gt; </span>                             |                       |         |                           |  |   |
| <b>VPN ID:</b> vpn-01b4b6197917b0aa2 <input type="button" value="X"/>   |                       |         |                           | <input type="button" value="Clear filters"/> |   |
| Name ▾  | VPN ID                | State   | Virtual private gateway ▾ | Tran   | ▼ |
| <input checked="" type="radio"/> mumbai-onprem-vpc  | vpn-01b4b6197917b0aa2 | Pending | vgw-030546fe44ba7bc7e     | -  |   |

#### 5. Downloading the Configuration file to be shared with the client.

### Download configuration

gateway. Please note these are samples, and will need modification to use Advanced Algorithms, Certificates, and/or IPv6.

**Vendor**  
The manufacturer of the customer gateway device (for example, Cisco Systems, Inc).

Generic

**Platform**  
The class of the customer gateway device (for example, J-Series).

Generic

**Software**  
The operating system running on the customer gateway device (for example, ScreenOS).

Vendor Agnostic

**IKE version**  
The IKE version you are using for your VPN connection.

ikev1

#### 6. Also enabled the route propagation in the Mumbai-vpc route table.

## Edit route propagation

### Route table basic details

Route table ID

 rtb-0b7cda26fa0da090d

### Edit route propagation

Virtual Private Gateway

[vgw-030546fe44ba7bc7e / vpg-1](#)

Propagation

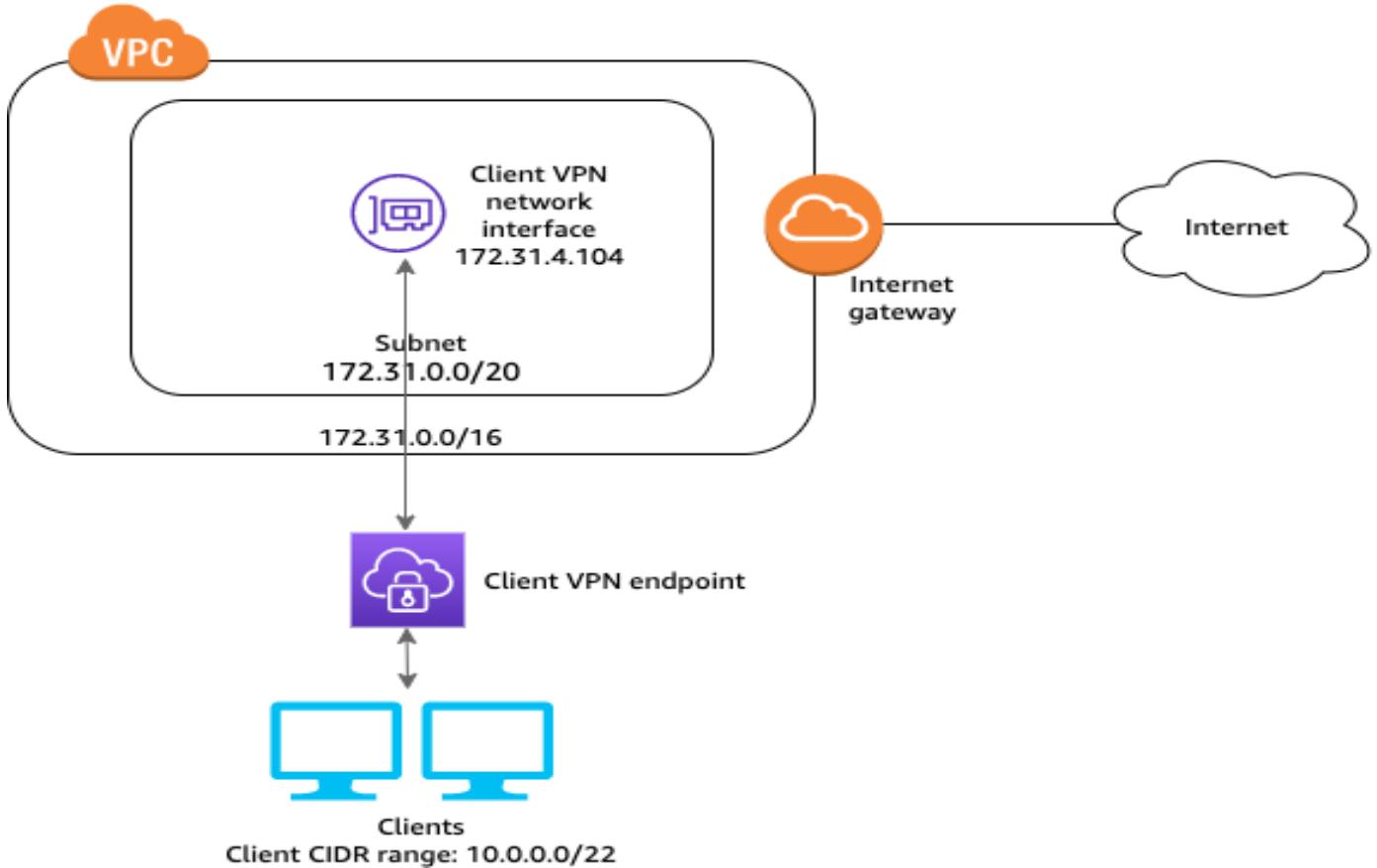
Enable

Cancel

Save

This completed the Site-to-site connectivity on AWS.

## 2. Set up Point to site connectivity on AWS.



1. Downloaded and installed open VPN connect.
2. Downloaded an installed Easy-RSA.
3. Renamed the extracted folder to EasyRSA3 then cut and pasted it in the folder of OpenVPN in Local Disk C:/ Program Files.
4. Opened Windows Terminal as Administrator and did the following to set up mutual authentication (server and client certificate).
  - a. Navigated to the location where Easy-RSA folder was pasted.

```
PS C:\WINDOWS\system32> cd 'C:\Program Files\OpenVPN'  
PS C:\Program Files\OpenVPN>
```

- b. Ran the following command on Command Prompt to activate the Easy-RSA Shell.

```
PS C:\Program Files\OpenVPN\EasyRSA> .\EasyRSA-Start.bat  
  
Welcome to the EasyRSA 3 Shell for Windows.  
Easy-RSA 3 is available under a GNU GPLv2 license.  
  
Invoke './easyrsa' to call the program. Without commands, help is displayed.  
  
EasyRSA Shell  
# |
```

- c. Initialized a new PKI environment.

```
# ./easyrsa init-pki
* Notice:

init-pki complete; you may now create a CA or requests.

Your newly created PKI dir is:
* C:/Program Files/OpenVPN/EasyRSA/pki

* Notice:
IMPORTANT: Easy-RSA 'vars' file has now been moved to your PKI above.
```

- d. Ran the following commands to build a new certificate authority (CA).
  - i. Ran the following command, specified common name as test.

```
If you enter '.', the field will be left blank.
```

```
-----
Common Name (eg: your user, host, or server name) [Easy-RSA CA]:
test
```

```
* Notice:
```

```
CA creation complete and you may now import and sign cert requests.
```

- ii. Generated server certificate and key.

```
EasyRSA Shell
# ./easyrsa build-server-full server nopass
* Notice:
Using Easy-RSA configuration from: C:/Program Files/OpenVPN/EasyRSA/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.3 3 May 2022 (Library: OpenSSL 3.0.3 3 May 20
22)
```

```
* Notice:
```

```
Keypair and certificate request completed. Your files are:
req: C:/Program Files/OpenVPN/EasyRSA/pki/reqs/server.req
key: C:/Program Files/OpenVPN/EasyRSA/pki/private/server.key
```

- iii. Generated client certificate and key.

```
EasyRSA Shell
# ./easyrsa build-client-full client1.domain.tld nopass
* Notice:
Using Easy-RSA configuration from: C:/Program Files/OpenVPN/EasyRSA/pki/vars

* Notice:
Using SSL: openssl OpenSSL 3.0.3 3 May 2022 (Library: OpenSSL 3.0.3 3 May 20
22)
```

\* Notice:

```
Keypair and certificate request completed. Your files are:  
req: C:/Program Files/OpenVPN/EasyRSA/pki/reqs/client1.domain.tld.req  
key: C:/Program Files/OpenVPN/EasyRSA/pki/private/client1.domain.tld.key
```

- iv. Exited the EasyRSA3 shell.
- e. Copied the server certificate and key and the client certificate and key to a custom folder by running the following commands.

```
PS C:\Program Files\OpenVPN\EasyRSA> mkdir C:\custom_folder
```

```
Directory: C:\  
  
Mode                LastWriteTime       Length Name  
----                -----          ---- -  
d----- 06-08-2022    20:21            custom_folder
```

```
PS C:\Program Files\OpenVPN\EasyRSA> copy pki\ca.crt C:\custom_folder  
PS C:\Program Files\OpenVPN\EasyRSA> copy pki\issued\server.crt C:\custom_folder  
PS C:\Program Files\OpenVPN\EasyRSA> copy pki\private\server.key C:\custom_folder  
PS C:\Program Files\OpenVPN\EasyRSA> copy pki\issued\client1.domain.tld.crt C:\custom_folder  
PS C:\Program Files\OpenVPN\EasyRSA> copy pki\private\client1.domain.tld.key C:\custom_folder  
PS C:\Program Files\OpenVPN\EasyRSA> cd C:\custom_folder|
```

We can see the generated certificates in the file explorer, inside the custom\_folder.

| Name                   | Date modified    | Type                 | Size |
|------------------------|------------------|----------------------|------|
| ca                     | 06-08-2022 20:14 | Security Certificate | 2 KB |
| client1.domain.tld     | 06-08-2022 20:18 | Security Certificate | 5 KB |
| client1.domain.tld.key | 06-08-2022 20:18 | KEY File             | 2 KB |
| server                 | 06-08-2022 20:17 | Security Certificate | 5 KB |
| server.key             | 06-08-2022 20:16 | KEY File             | 2 KB |

- f. Generated an IAM user for point-to-site connectivity and allowed it Administrator Access, then configured the user inside the custom folder.
- g. Uploaded the server certificate and key and client certificate and key to ACM using the following commands.

```
PS C:\custom_folder> aws acm import-certificate --certificate file://server.crt --private-key file://server.key --certificate-chain file://ca.crt  
{  
    "CertificateArn": "arn:aws:acm:ap-south-1:258046353232:certificate/fb0ffd4e-5367-4960-8a4a-c5aa0e086e86"  
}  
  
PS C:\custom_folder>
```

```

PS C:\custom_folder> aws acm import-certificate --certificate fileb://client1.domain.tld.crt --private-key fileb://client1.domain.tld.key --certificate-chain fileb://ca.crt
{
    "CertificateArn": "arn:aws:acm:ap-south-1:258046353232:certificate/4390e702-4777-48b0-a920-6076c3bf572d"
}

PS C:\custom_folder>

```

We can now see the certificates in the ACM section in AWS console.

| AWS Certificate Manager  | >                                    | Certificates       |          |                     |         |                     |
|--------------------------|--------------------------------------|--------------------|----------|---------------------|---------|---------------------|
| Certificates (3)         |                                      |                    |          |                     |         |                     |
|                          | Certificate ID                       | Domain name        | Type     | Status              | In use? | Renewal eligibility |
| <input type="checkbox"/> | 4390e702-4777-48b0-a920-6076c3bf572d | client1.domain.tld | Imported | <span>Issued</span> | No      | Ineligible          |
| <input type="checkbox"/> | fb0ffd4e-5367-4960-8a4a-c5aa0e086e86 | server             | Imported | <span>Issued</span> | No      | Ineligible          |

- Created a client VPN endpoint using AWS console (while configuring make sure to enable split tunnelling).

The screenshot shows the AWS Client VPN endpoints page. It displays a table with one row for the endpoint 'client-vpn-endpoint-1'. The columns are: Name, Client VPN endpoint ID, State, and Client CIDR. The endpoint details are shown in a modal window below the table.

| Name                  | Client VPN endpoint ID          | State             | Client CIDR |
|-----------------------|---------------------------------|-------------------|-------------|
| client-vpn-endpoint-1 | cvpn-endpoint-03261dc63d6282fbc | Pending-associate | 11.0.0.0/16 |

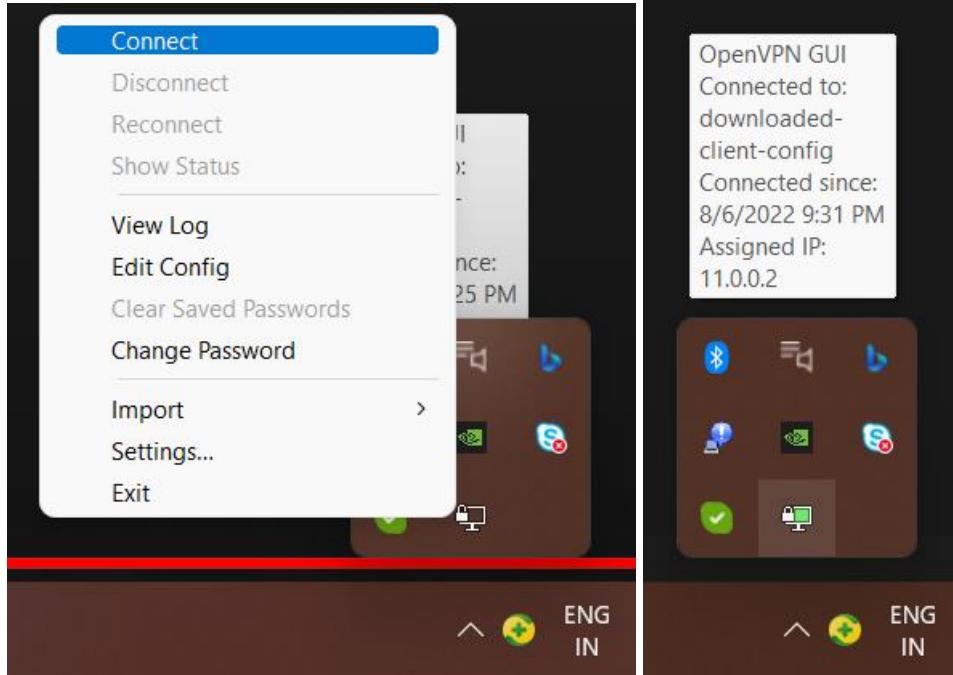
**client-vpn-endpoint-03261dc63d6282fbc / client-vpn-endpoint-1**

**Details**

|   |  |                            |                           |
|---|--|----------------------------|---------------------------|
| Client VPN endpoint ID<br>cvpn-endpoint-03261dc63d6282fbc | Server certificate ARN<br>arn:aws:acm:ap-south-1:258046353232:certificate/fb0ffd4e-5367-4960-8a4a-c5aa0e086e86 | Connection log<br>false    | Transport protocol<br>udp |
| Description<br>client-vpn-endpoint-1                      | Cloudwatch log group<br>-  | Cloudwatch log stream<br>- | Split-tunnel<br>Enabled   |
| State<br>Pending-associate                                | Creation time<br>August 6, 2022, 03:30 (UTC+05:30)   | Cloudwatch log stream<br>- | VPC ID<br>-               |
| Authentication type                                       | VPN port<br>11.0.0.16  | Client CIDR                | Self-service portal URL   |

- Associated a target network to Mumbai-VPC.
- Added an authorization rule to the Client VPN endpoint.

8. While the client VPN endpoint was configured, I launched an EC2 instance inside the Mumbai-VPC, I did not provide a public IP for connecting to the Instance.
9. When the client-VPN-endpoint became active, I downloaded the client configuration file and performed the following steps.
  - a. Opened the configuration file using notepad and inserted the following at the end.
  - b. Inserted the client certificate file's path.
  - c. Inserted the client key file's path.
10. Saved the file and moved the configuration file to config folder inside OpenVPN folder.
11. Connected to the server using OpenVPN client GUI present in hidden icons in the taskbar.



12. After connection was successful, I connected to the EC2 instance using its private IP from my computer using xshell and was successful.

```
Xshell 7 (Build 0111)
Copyright (c) 2020 NetSarang Computer, Inc. All rights reserved.

Type `help' to learn how to use Xshell prompt.
[C:\~]$ ssh -i "VPC-server-sumit.pem" ec2-user@7.0.1.40

Connecting to 7.0.1.40:22...
Connection established.
To escape to local shell, press 'Ctrl+Alt+]'.

WARNING! The remote SSH server rejected X11 forwarding request.

 _|_(_/_ )   Amazon Linux 2 AMI
__|\_\_|_|
https://aws.amazon.com/amazon-linux-2/
[ec2-user@ip-7-0-1-40 ~]$
```

13. So, I can say that I successfully configured Point-to-Site connectivity using AWS.

### 3. Transit Gateway setup.

1. Created 3 VPCs in the Mumbai region.

| Your VPCs (3) <a href="#">Info</a> |              |                       |           |            |           |  | <a href="#">Actions</a> |
|------------------------------------|--------------|-----------------------|-----------|------------|-----------|--|-------------------------|
| <input type="text"/> Filter VPCs   |              |                       |           |            |           |  |                         |
| <input type="checkbox"/>           | Name         | VPC ID                | State     | IPv4 CIDR  | IPv6 CIDR |  | <a href="#">Actions</a> |
| <input type="checkbox"/>           | mumbai-vpc-3 | vpc-0c6f3fce45ee39001 | Available | 9.0.0.0/16 | -         |  | <a href="#">Actions</a> |
| <input type="checkbox"/>           | mumbai-vpc-1 | vpc-083ef3fe083e0f251 | Available | 7.0.0.0/16 | -         |  | <a href="#">Actions</a> |
| <input type="checkbox"/>           | mumbai-vpc-2 | vpc-0996c0d7154259fe0 | Available | 8.0.0.0/16 | -         |  | <a href="#">Actions</a> |

2. Created subnets in the 3 VPCs.

| Subnets (4) <a href="#">Info</a>    |                    |                          |           |                               |            |  | <a href="#">Actions</a> |
|-------------------------------------|--------------------|--------------------------|-----------|-------------------------------|------------|--|-------------------------|
| <input type="text"/> Filter subnets |                    |                          |           |                               |            |  |                         |
| <input type="checkbox"/>            | Name               | Subnet ID                | State     | VPC                           | IPv4 CIDR  |  | <a href="#">Actions</a> |
| <input type="checkbox"/>            | vpn-3-subnet       | subnet-0215dd04eadb2a6bd | Available | vpc-0c6f3fce45ee39001   mu... | 9.0.1.0/24 |  | <a href="#">Actions</a> |
| <input type="checkbox"/>            | vpn-2-subnet       | subnet-0a2617e69088d06bf | Available | vpc-0996c0d7154259fe0   mu... | 8.0.1.0/24 |  | <a href="#">Actions</a> |
| <input type="checkbox"/>            | vpn-1-web-subnet-2 | subnet-0225018fd4692efec | Available | vpc-083ef3fe083e0f251   mu... | 7.0.2.0/24 |  | <a href="#">Actions</a> |
| <input type="checkbox"/>            | vpn-1-web-subnet-1 | subnet-0c89d2c7eab4ba5ac | Available | vpc-083ef3fe083e0f251   mu... | 7.0.1.0/24 |  | <a href="#">Actions</a> |

3. Created route tables for the 3 VPCs and associated them with their respective subnets.

| Route tables (4) <a href="#">Info</a>    |                   |                       |                             |                   |      |                       | <a href="#">Actions</a> |
|--|-------------------|-----------------------|-----------------------------|-------------------|------|-----------------------|-------------------------|
| <input type="text"/> Filter route tables |                   |                       |                             |                   |      |                       |                         |
| <input type="checkbox"/>                 | Name              | Route table ID        | Explicit subnet associat... | Edge associations | Main | VPC                   | <a href="#">Actions</a> |
| <input type="checkbox"/>                 | vpn-1-subnet-1-rt | rtb-0b7cda26fa0da090d | subnet-0c89d2c7eab4b...     | -                 | No   | vpc-083ef3fe083e0f251 | <a href="#">Actions</a> |
| <input type="checkbox"/>                 | vpn-1-subnet-2-rt | rtb-099267a3f5edfd01  | subnet-0225018fd4692...     | -                 | No   | vpc-083ef3fe083e0f251 | <a href="#">Actions</a> |
| <input type="checkbox"/>                 | vpn-3-rt          | rtb-0aefcc7163e979171 | subnet-0215dd04eadb2...     | -                 | No   | vpc-0c6f3fce45ee39001 | <a href="#">Actions</a> |
| <input type="checkbox"/>                 | vpn-2-rt          | rtb-065a2b4e130b376ae | subnet-0a2617e69088d...     | -                 | No   | vpc-0996c0d7154259fe0 | <a href="#">Actions</a> |

4. Created internet gateways for the 3 VPCs, attached them to their respective VPCs and added the respective internet gateways to the respective route tables.

| Internet gateways (3) <a href="#">Info</a>    |           |                       |          |                                      |              |  | <a href="#">Actions</a> | <a href="#">Create Internet gateway</a> |
|---|-----------|-----------------------|----------|--------------------------------------|--------------|--|-------------------------|---|
| <input type="text"/> Filter internet gateways |           |                       |          |                                      |              |  |                         |   |
| <input type="checkbox"/>                      | Name      | Internet gateway ID   | State    | VPC ID                               | Owner        |  | <a href="#">Actions</a> |   |
| <input type="checkbox"/>                      | vpn-2-igw | igw-0339049dca1b323e1 | Attached | vpc-0996c0d7154259fe0   mumbai-vpc-2 | 258046353232 |  | <a href="#">Actions</a> |   |
| <input type="checkbox"/>                      | vpn-3-igw | igw-0bcdd4be4ba18504a | Attached | vpc-0c6f3fce45ee39001   mumbai-vpc-3 | 258046353232 |  | <a href="#">Actions</a> |   |
| <input type="checkbox"/>                      | vpn-1-igw | igw-0f10d2feb4472a074 | Attached | vpc-083ef3fe083e0f251   mumbai-vpc-1 | 258046353232 |  | <a href="#">Actions</a> |   |

5. Created an instance inside each VPC, enabling public IP access to instance inside VPC-1 only.

| Instances (3) <a href="#">Info</a> |              | <a href="#">Connect</a> | Instance state       | Actions       | Launch instances               |              |                   |             |    |
|------------------------------------|--------------|-------------------------|----------------------|---------------|--------------------------------|--------------|-------------------|-------------|----|
|                                    | Name         | Instance ID             | Instance state       | Instance type | Status check                   | Alarm status | Availability Zone | Public IPv4 | DH |
| <input type="checkbox"/>           | vpc-1-server | i-077cf725cd8e4e932     | <span>Running</span> | t2.micro      | <span>2/2 checks passed</span> | No alarms    | ap-south-1a       | -           |    |
| <input type="checkbox"/>           | vpc-2-server | i-0f245717dd3d42948     | <span>Running</span> | t2.micro      | <span>2/2 checks passed</span> | No alarms    | ap-south-1b       | -           |    |
| <input type="checkbox"/>           | vpc-3-server | i-0c9a7727772e6d52e     | <span>Running</span> | t2.micro      | <span>2/2 checks passed</span> | No alarms    | ap-south-1b       | -           |    |

- a. Now I connected to the instance inside Mumbai-vpc-1 and tried to ping the other two instances, but it failed.

```
[root@ip-7-0-1-38 ~]# ping 8.0.1.227
PING 8.0.1.227 (8.0.1.227) 56(84) bytes of data.
^C
--- 8.0.1.227 ping statistics ---
4 packets transmitted, 0 received, 100% packet loss, time 3052ms

[root@ip-7-0-1-38 ~]# ping 9.0.1.98
PING 9.0.1.98 (9.0.1.98) 56(84) bytes of data.
^C
--- 9.0.1.98 ping statistics ---
3 packets transmitted, 0 received, 100% packet loss, time 2048ms
```

6. Created a transit gateway and used it to create routes in each route table of the VPCs in order to access the other VPCs.
- a. Created a transit gateway.

| Transit gateways (1/1) <a href="#">Info</a> |                       |                       |              |                        |
|---|-----------------------|-----------------------|--------------|------------------------|
| <input checked="" type="checkbox"/>         | Name                  | Transit gateway ID    | Owner ID     | State                  |
| <input checked="" type="checkbox"/>         | transit-gateway-mu... | tgw-01f6afece0f9f14a8 | 258046353232 | <span>Available</span> |

- b. Created transit gateway attachments for each VPC.

| Transit gateway attachments (3) <a href="#">Info</a> |                  |                               |                       |               |                       |                        |           |  |
|--|------------------|-------------------------------|-----------------------|---------------|-----------------------|------------------------|-----------|--|
| <input type="checkbox"/>                             | Name             | Transit gateway attachment ID | Transit gateway ID    | Resource type | Resource ID           | State                  | Associati |  |
| <input type="checkbox"/>                             | tgw-attach-vpc-3 | tgw-attach-02ceb9b47f1ad914c  | tgw-01f6afece0f9f14a8 | VPC           | vpc-0c6f3fce45ee39001 | <span>Available</span> | tgw-rtb-C |  |
| <input type="checkbox"/>                             | tgw-attach-vpc-2 | tgw-attach-03eabb5036a5640ef  | tgw-01f6afece0f9f14a8 | VPC           | vpc-0996c0d7154259fe0 | <span>Available</span> | tgw-rtb-C |  |
| <input type="checkbox"/>                             | tgw-attach-vpc-1 | tgw-attach-0a5f4f0c055009ec0  | tgw-01f6afece0f9f14a8 | VPC           | vpc-083ef3fe083e0f251 | <span>Available</span> | tgw-rtb-C |  |

- c. Attachments were attached to the individual route tables of the VPCs.
- d. Now, I tried to ping the other two instances using their private IP, it successfully pinged the information.

```
[root@ip-7-0-1-38 ~]# ping 9.0.1.98
PING 9.0.1.98 (9.0.1.98) 56(84) bytes of data.
64 bytes from 9.0.1.98: icmp_seq=1 ttl=254 time=1.56 ms
64 bytes from 9.0.1.98: icmp_seq=2 ttl=254 time=1.26 ms
^C
--- 9.0.1.98 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 1.262/1.413/1.564/0.151 ms
[root@ip-7-0-1-38 ~]# ping 8.0.1.227
PING 8.0.1.227 (8.0.1.227) 56(84) bytes of data.
64 bytes from 8.0.1.227: icmp_seq=1 ttl=254 time=1.63 ms
64 bytes from 8.0.1.227: icmp_seq=2 ttl=254 time=0.928 ms
^C
--- 8.0.1.227 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.928/1.281/1.634/0.353 ms
[root@ip-7-0-1-38 ~]#
```

## 7. Created a new VPN attachment for site-to-site connectivity purpose.

| Transit gateway attachments (1) <a href="#">Info</a>  |                               |                       |               |                       |  |          |
|---|-------------------------------|-----------------------|---------------|-----------------------|--|----------|
| <a href="#">Filter transit gateway attachments</a> <span style="float: right;">Actions ▾ <a href="#">Create transit gateway attachment</a></span> |                               |                       |               |                       |  |          |
| Name  | Transit gateway attachment ID | Transit gateway ID    | Resource type | Resource ID           | State  | Associat |
| tgw-vpn   | tgw-attach-05164e97d2dd63ca1  | tgw-01f6afece0f9f14a8 | VPN           | vpn-02c286c6b460f47e4 | <span style="color: green;">Available</span> | tgw-rtb- |

- a. In the backend AWS will create a customer gateway and site-to-site gateway and enable routing.

| Customer gateways (1/1) <a href="#">Info</a>  |                       |  |         |              |         |  |
|---|-----------------------|--|---------|--------------|---------|--|
| <a href="#">Filter customer gateways</a> <span style="float: right;">Actions ▾ <a href="#">Create customer gateway</a></span> |                       |  |         |              |         |  |
| Name  | Customer gateway ID   | State  | BGP ASN | IP address   | Type    |  |
| -   | cgw-0f3fb2c2c185309bb | <span style="color: green;">Available</span> | 65000   | 5.59.110.159 | ipsec.1 |  |

| VPN connections (1/1) <a href="#">Info</a>   |                       |  |                         |                       |                       |  |
|--|-----------------------|--|-------------------------|-----------------------|-----------------------|--|
| <a href="#">Filter VPN connections</a> <span style="float: right;">Actions ▾ <a href="#">Download configuration</a> <a href="#">Create VPN connection</a></span> |                       |  |                         |                       |                       |  |
| Name   | VPN ID                | State  | Virtual private gateway | Transit gateway       | Customer gate         |  |
| -  | vpn-02c286c6b460f47e4 | <span style="color: green;">Available</span> | -                       | tgw-01f6afece0f9f14a8 | cgw-0f3fb2c2c185309bb |  |

- b. We have to download the client configuration file and give it to the client side engineers to configure the connection.
- c. We can do some CIDR changes in the transit gateway routing table for site-to-site connectivity.

## 8. Transit Gateway set up was configured for the 3 VPCs successfully.



8/5/2022

# Project - 04

Migrate a database server from on-premises to AWS.



Sumit Mishra  
SIC: 190310286

## 1. Create a RDS server on AWS using the console.

The screenshot shows the AWS RDS Databases console. At the top, there's a navigation bar with 'RDS' and 'Databases'. Below it is a search bar labeled 'Filter by databases'. The main area is a table titled 'Databases' with columns: DB identifier, Role, Engine, Region & AZ, Size, Status, CPU, and Cu. A single row is visible for 'database-1', which is an Instance of SQL Server Standard Edition in the ap-south-1c region, db.m5.xlarge size, and status 'Configuring-enhanced-monitoring'.

## 2. Create an EC2 instance (Amazon Linux with ms SQL server) using the console.

The screenshot shows the AWS Instances (1/4) console. It lists four instances: 'vpc-1-server', 'db-server', 'vpc-2-server', and 'vpc-3-server'. The 'db-server' instance is selected, indicated by a checked checkbox. The table has columns: Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4. The 'db-server' instance is in the 'Running' state, t3.xlarge type, with 2/2 checks passed, no alarms, in the ap-south-1c zone, and has a public IP ec2-3-6-7-229.

## 3. Configure SQL Server on the EC2 instance.

1. While the server was launching, I created and attached an extra EBS volume to store the Database data.
2. Connected to the EC2 instance using xShell and typed the following commands for configuring the SQL server.
  - a. Switched user to super user.
  - b. Stopped the SQL server.
  - c. Ran an mssql-conf script to reset the SA password.
  - d. Formatted the volume to ext4 type.
  - e. Created a directory name 'SQLServerData'.
  - f. Mounted the directory into the attached volume.
  - g. Attached some permissions to the directory.
  - h. Changed some settings using some commands to enable the SQLServerData director as the default data directory.
  - i. Restarted the ms-sql service.
3. Connected the ms-sql server using the Microsoft SQL Server Management Studio.
  - a. Provide username as sa.
  - b. Provide password for the system server.
  - c. Select SQL server authentication.
  - d. Click connect.
4. Created a database and some schema inside using the Microsoft SQL Server Management Studio.
  - a. Created a database using 'Create Database' Command.
  - b. Created a sample table inside the database.

```
CREATE TABLE Persons (
    PersonID int,
    LastName varchar(255),
    FirstName varchar(255),
    Address varchar(255),
    City varchar(255)
);
```

- c. Inserted some records into it.

```
INSERT INTO Persons(PersonID, LastName, FirstName, Address, City)
VALUES
(1, 'Mishra', 'Sumit', 'Barmunda', 'BBSR'),
(2, 'Mishra', 'Amit', 'Barmunda', 'BBSR'),
(3, 'Mishra', 'Subrat', 'Barmunda', 'BBSR'),
(4, 'Mishra', 'Mamata', 'Barmunda', 'BBSR'),
(5, 'Mishra', 'Nabneet', 'Barmunda', 'BBSR');
```

## 5. Migrating the on-premises database to AWS using DMS service.

1. Created a replication instance for migration of Databases using port number 1433 for mssql.

The screenshot shows the AWS DMS Replication Instances page. At the top, there is a message about upgrading to version 3.4.7 and higher. Below that is a table titled 'Replication instances (1)'. The table has columns for Name, Status, VPC, Class, Engine version, Availability zone, Public, and Public. The single entry is 'sql-replication-instance' with a status of 'Creating'.

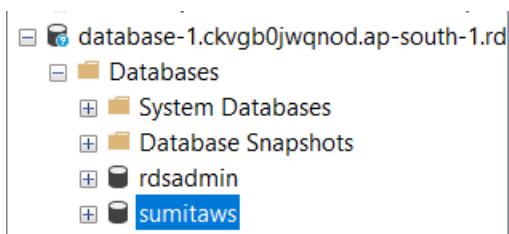
| Name                     | Status   | VPC                   | Class         | Engine version | Availability zone | Public | Public |
|--------------------------|----------|-----------------------|---------------|----------------|-------------------|--------|--------|
| sql-replication-instance | Creating | vpc-01d987e17c3785149 | dms.t3.medium | 3.4.7          | ap-south-1c       | Yes    | Yes    |

2. Created a source server endpoint using the source server IP (Linux server's public IP) and tested it.

The screenshot shows the 'Test endpoint connection' page. It displays a table with one row for 'linux-sql-server' with a status of 'successful'. A 'Run test' button is visible above the table.

| Endpoint identifier | Replication instance     | Status     | Message |
|---------------------|--------------------------|------------|---------|
| linux-sql-server    | sql-replication-instance | successful |         |

3. Connected to the RDS from Microsoft SQL Server Management Studio and created a database inside it.



4. Created the destination server endpoint using the RDS and tested it.

**Run test**

| Endpoint identifier | Replication instance     | Status     | Message |
|---------------------|--------------------------|------------|---------|
| database-1          | sql-replication-instance | successful |         |

Cancel

**Create endpoint**

5. Create a database migration task.

- Selected the created replication instance.
- Selected the created source server endpoint.
- Selected the created destination endpoint.
- Assigned % in selection rules to enable full data migration.
- Added tags and created the task.

DMS > Database migration tasks

**Database migration tasks (1)**

| Identifier           | Status   | Progress  | Type             | Source     | Target                   | Replication instance | Started | Stopped |
|----------------------|----------|-----------|------------------|------------|--------------------------|----------------------|---------|---------|
| sql-replication-task | Creating | Full load | linux-sql-server | database-1 | sql-replication-instance | -                    | -       | -       |

6. Waited for the migration task to fully complete, then tested using the Microsoft SQL Server management Studio to check if the contents of the on-premises server have been fully transferred into the RDS server on AWS.

DMS > Database migration tasks

**Database migration tasks (1)**

| Identifier           | Status        | Progress | Type      | Source           | Target     | Replication instance     | Started               |
|----------------------|---------------|----------|-----------|------------------|------------|--------------------------|-----------------------|
| sql-replication-task | Load complete | 100%     | Full load | linux-sql-server | database-1 | sql-replication-instance | August 7, 2022 at 14: |

```
select * from Persons;
```

380 %

Results Messages

|   | PersonID | LastName | FirstName | Address  | City |
|---|----------|----------|-----------|----------|------|
| 1 | 1        | Mishra   | Sumit     | Barmunda | BBSR |
| 2 | 2        | Mishra   | Amit      | Barmunda | BBSR |
| 3 | 3        | Mishra   | Subrat    | Barmunda | BBSR |
| 4 | 4        | Mishra   | Mamata    | Barmunda | BBSR |
| 5 | 5        | Mishra   | Nabneet   | Barmunda | BBSR |

Using the above mentioned steps, I successfully completed the Database Migration Task from on-premises server to AWS.



7/23/2022

# Project-05

Mount S3 bucket as a drive in both linux and windows, configure MFA delete and versioning on the bucket.

Sumit Mishra  
190310286

## 1. Mount an S3 bucket on ubuntu AMI.

### 1. Created an EC2 instance to mount an S3 bucket using the Ubuntu AMI available.

The screenshot shows the AWS CloudWatch Metrics interface. A metric named 'CPU Utilization' is selected. The chart displays data over a period of 1 hour, showing a significant spike in CPU usage starting around July 23, 2022, at 16:17:03 UTC. The Y-axis represents CPU Utilization from 0% to 100%, and the X-axis shows time intervals. The peak value is approximately 95%.

### 2. Created an S3 bucket to mount to the EC2 instance. (Also uploaded a file named 'abc.text').

The screenshot shows the AWS S3 Buckets page. A bucket named 's3bucketofubuntu' is selected. The 'Objects' tab is active, showing one object named 'abc.txt'. The object details indicate it was uploaded on July 23, 2022, at 16:17:03 UTC, is 0 bytes in size, and has a storage class of Standard. The object's type is listed as 'txt'.

### 3. Connected to the EC2 instance using SSH client.

### 4. Fired the following commands to setup and mount the s3 bucket on the instance:

a) 'sudo apt-get update' to update all libraries of Ubuntu.

```
root@ip-172-31-42-167:~# sudo apt-get update
Hit:1 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy InRelease
Get:2 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-updates InRelease [114 kB]
Get:3 http://security.ubuntu.com/ubuntu jammy-security InRelease [110 kB]
Get:4 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy-backports InRelease [99.8 kB]
Get:5 http://ap-south-1.ec2.archive.ubuntu.com/ubuntu jammy/universe amd64 Packages [14.1 MB]
Get:6 http://security.ubuntu.com/ubuntu jammy-security/main amd64 Packages [228 kB]
Get:7 http://security.ubuntu.com/ubuntu jammy-security/main Translation-en [56.1 kB]
Get:8 http://security.ubuntu.com/ubuntu jammy-security/main amd64 c-n-f Metadata [3564 B]
Get:9 http://security.ubuntu.com/ubuntu jammy-security/restricted amd64 Packages [203 kB]
```

b) 'sudo apt-get install s3fs' to install s3fs.

```
root@ip-172-31-42-167:~# sudo apt-get install s3fs
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  bzip2 libfuse2 mailcap mime-support
Suggested packages:
  bzip2-doc
The following NEW packages will be installed:
```

c) 'echo

AKIATYFGC35IFYZNP7HA:5WR1bzCxVytPcfp1zRDedFJnR/JLD8Rm3zHqUxcP>~/.passwd-s3fs' to create a configuration file for IAM user having access key and secret access key.

```
root@ip-172-31-42-167:~# echo AKIATYFGC35IFYZNP7HA:5WR1bzCxVytPcfp1zRDedFJnR/JLD8Rm3zHqUxcP>~/.passwd-s3fs
```

d) 'cat ~/.passwd-s3fs' to check if the file has been created and contents have been filled in it or not.

```
root@ip-172-31-42-167:~# cat ~/.passwd-s3fs
```

e) 'chmod 600 ~/.passwd-s3fs' to set permissions for the configuration file.

```
root@ip-172-31-42-167:~# chmod 600 `/.passwd-s3fs`
```

f) 'mkdir ~/s3-bucket-folder' to make a directory to mount the s3 volume to.

```
root@ip-172-31-42-167:~# chmod 600 `/.passwd-s3fs`
root@ip-172-31-42-167:~# mkdir ~/s3-bucket-folder
```

g) 's3fs s3bucketofubuntu ~/s3-bucket-folder -o passwd\_file=~/./passwd-s3fs' to mount the s3 bucket to the specified directory using the specified configuration file.

h) 'mount', 'df -h' to check if the s3 bucket has been mounted to the directory or not.

```
root@ip-172-31-42-167:~# s3fs s3bucketofubuntu ~/s3-bucket-folder -o passwd_file=~/./passwd-s3fs
root@ip-172-31-42-167:~# mount
```

i) 'cd ~/s3-bucket-folder' to go the directory where the volume has been mounted.

j) 'ls' to check if the file 'abc.txt' is present or not.

k) 'touch xyz.txt' to create a text inside that directory and check if it reflects in the s3 bucket in the console or not.

| Objects (2)  |         |      |                                     |      |               |                     |
|--|---------|------|-------------------------------------|------|---------------|---------------------|
| Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory <a href="#">[?]</a> to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more <a href="#">[?]</a> |         |      |                                     |      |               |                     |
|  | Name    | Type | Last modified                       | Size | Storage class |                     |
| <input type="checkbox"/>   | abc.txt | txt  | July 23, 2022, 16:17:03 (UTC+05:30) | 0 B  | Standard      | <a href="#">[?]</a> |
| <input type="checkbox"/>   | xyz.txt | txt  | July 23, 2022, 18:04:49 (UTC+05:30) | 0 B  | Standard      | <a href="#">[?]</a> |

## 2. Enable MFA delete on s3 bucket.

### 1. Creating a new access key and secret access key for the root user account.

▼ Access keys (access key ID and secret access key)

Use access keys to make programmatic calls to AWS from the AWS CLI, Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time.

For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation.  
If you lose or forget your secret key, you cannot retrieve it. Instead, create a new access key and make the old key inactive. Learn more

| Created       | Access Key ID        | Last Used | Last Used Region | Last Used Service | Status | Actions  |
|---------------|----------------------|-----------|------------------|-------------------|--------|--|
| Jul 23rd 2022 | AKIATYFGC35IAUXPXE65 | N/A       | N/A              | N/A               | Active | <a href="#">Make Inactive</a>   <a href="#">Delete</a> |

[Create New Access Key](#)

### 2. Configuring IAM account with the help of the generated access key and secret access key of the root user account.

```
PS C:\Users\cr7su> aws configure
AWS Access Key ID [*****P7HA]: AKIATYFGC35IAUXPXE65
AWS Secret Access Key [*****UxcP]: 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
Default region name [ap-south-1]:
Default output format [json]:
PS C:\Users\cr7su> |
```

### 3. 'aws s3 ls' to display all the buckets in this region.

```
PS C:\Users\cr7su> aws s3 ls
2022-07-23 16:16:53 s3bucketofubuntu
```

### 4. 'aws s3api put-bucket-versioning --bucket s3bucketofubuntu --versioning-configuration Status=Enabled,MFADelete=Enabled --mfa "arn:aws:iam::258046353232:mfa/root-account-mfa-device 090201"' to enable MFA delete on the root account for s3 bucket.

```
PS C:\Users\cr7su> aws s3api put-bucket-versioning --bucket s3bucketofubuntu --versioning-configuration Status=Enabled,MFADelete=Enabled --mfa "arn:aws:iam::258046353232:mfa/root-account-mfa-device 090201"
```

### 5. Checking if MFA delete and bucket versioning for the s3 bucket are enabled or not (also reflected in the console).

```
PS C:\Users\cr7su> aws s3api get-bucket-versioning --bucket s3bucketofubuntu
{
    "Status": "Enabled",
    "MFADelete": "Enabled"
}
```

**Bucket Versioning**

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

[Edit](#)

Bucket Versioning  
Enabled

Multi-factor authentication (MFA) delete  
An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Enabled

## 6. Deleting the 729 B abc.txt file without providing MFA key using CLI command.

```
PS C:\Users\cr7su> aws s3api delete-object --bucket s3bucketofubuntu --key abc.txt --version-id _4Ck2K8yjCwOdNf_AwntIS7qUI08GTDe
An error occurred (AccessDenied) when calling the DeleteObject operation: Mfa Authentication must be used for this request
PS C:\Users\cr7su> |
```

## 7. Now, trying to delete the same file version using MFA code using CLI command. (We can see that the version of the abc.txt file has been deleted).

```
PS C:\Users\cr7su> aws s3api delete-object --bucket s3bucketofubuntu --key abc.txt --version-id _4Ck2K8yjCwOdNf_AwntIS7qUI08GTDe --mfa "arn:aws:iam::258046353232:mfa/root-account-mfa-device 399091"
{
    "VersionId": "_4Ck2K8yjCwOdNf_AwntIS7qUI08GTDe"
}
```

## 8. Now, trying to delete xyz.txt version using the s3 management console we get error message.

Amazon S3 > Buckets > s3bucketofubuntu > Delete objects

**Delete objects** Info

✖ You can't delete object versions because Multi-factor authentication (MFA) delete is enabled for this bucket.  
To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

[Cancel](#)

## 3. Syncing files between two s3 buckets.

### 1. Creating Source bucket and enabling versioning to it.

```
PS C:\Users\cr7su> aws s3api create-bucket --bucket source-bucket-for-sumit
--region ap-south-1 --create-bucket-configuration LocationConstraint=ap-south-1
{
    "Location": "http://source-bucket-for-sumit.s3.amazonaws.com/"
}

PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su>
PS C:\Users\cr7su> aws s3api put-bucket-versioning --bucket source-bucket-for-sumit --versioning-configuration Status=Enabled
PS C:\Users\cr7su> |
```

## 2. Creating the destination bucket and enabling versioning to it.

```
PS C:\Users\cr7su> aws s3api create-bucket --bucket destn-bucket-for-sumit --region ap-south-1 --create-bucket-configuration LocationConstraint=ap-south-1
{
    "Location": "http://destn-bucket-for-sumit.s3.amazonaws.com/"
}

PS C:\Users\cr7su> aws s3api put-bucket-versioning --bucket destn-bucket-for-sumit --versioning-configuration Status=Enabled
PS C:\Users\cr7su> |
```

## 3. Uploading some files into the source bucket.

Amazon S3 > Buckets > source-bucket-for-sumit

source-bucket-for-sumit [Info](#)

[Objects](#) [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

**Objects (2)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

| <input type="checkbox"/> | Name                     | Type | Last modified                       | Size     | Storage class |
|--------------------------|--------------------------|------|-------------------------------------|----------|---------------|
| <input type="checkbox"/> | feelings.txt             | txt  | July 23, 2022, 21:34:45 (UTC+05:30) | 740.0 B  | Standard      |
| <input type="checkbox"/> | Sumit Mishra_Resume.docx | docx | July 23, 2022, 21:35:21 (UTC+05:30) | 231.7 KB | Standard      |

## 4. Syncing the files from source bucket to destination bucket using CLI.

```
PS C:\Users\cr7su> aws s3 sync s3://source-bucket-for-sumit/ s3://destn-bucket-for-sumit/
PS C:\Users\cr7su> |
```

## 5. Checking if the synced files are being reflected in the destination bucket in the s3 management console.

Amazon S3 > Buckets > destn-bucket-for-sumit

### destn-bucket-for-sumit [Info](#)

**Objects** [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

**Objects (2)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

| <input type="checkbox"/> | Name                     | Type | Last modified                       | Size     | Storage class |
|--------------------------|--------------------------|------|-------------------------------------|----------|---------------|
| <input type="checkbox"/> | feelings.txt             | txt  | July 23, 2022, 21:34:45 (UTC+05:30) | 740.0 B  | Standard      |
| <input type="checkbox"/> | Sumit Mishra_Resume.docx | docx | July 23, 2022, 21:35:21 (UTC+05:30) | 231.7 KB | Standard      |

## 4. Mounting an s3 bucket on windows.

1. Create an S3 bucket on AWS console and upload some files into it.
2. Make them public using ACL.

Amazon S3 > Buckets > s3bucketforsumit

### s3bucketforsumit [Info](#)

**Objects** [Properties](#) [Permissions](#) [Metrics](#) [Management](#) [Access Points](#)

**Objects (3)**

Objects are the fundamental entities stored in Amazon S3. You can use [Amazon S3 inventory](#) to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. [Learn more](#)

| <input checked="" type="checkbox"/> | Name            | Type   | Last modified                        | Size    | Storage class |
|-------------------------------------|-----------------|--------|--------------------------------------|---------|---------------|
| <input checked="" type="checkbox"/> | assets/         | Folder | -                                    | -       | -             |
| <input checked="" type="checkbox"/> | components.html | html   | August 1, 2022, 13:07:01 (UTC+05:30) | 40.2 KB | Standard      |
| <input checked="" type="checkbox"/> | index.html      | html   | August 1, 2022, 13:07:02 (UTC+05:30) | 47.4 KB | Standard      |

3. Allow versioning to the bucket.

## Edit Bucket Versioning [Info](#)

### Bucket Versioning

Versioning is a means of keeping multiple variants of an object in the same bucket. You can use versioning to preserve, retrieve, and restore every version of every object stored in your Amazon S3 bucket. With versioning, you can easily recover from both unintended user actions and application failures. [Learn more](#)

#### Bucket Versioning

Suspend

This suspends the creation of object versions for all operations but preserves any existing object versions.

Enable

#### Multi-factor authentication (MFA) delete

An additional layer of security that requires multi-factor authentication for changing Bucket Versioning settings and permanently deleting object versions. To modify MFA delete settings, use the AWS CLI, AWS SDK, or the Amazon S3 REST API. [Learn more](#)

Disabled

[Cancel](#)

[Save changes](#)

4. Download tntdrive from the internet and install it.

5. Create an IAM user for accessing the s3 bucket.

### Users (1) [Info](#)

An IAM user is an identity with long-term credentials that is used to interact with AWS in an account.

[Delete](#)

[Add users](#)

Find users by username or access key

< 1 >

| <input type="checkbox"/> | User name | Groups | Last activity | MFA  | Password age | Active key age |
|--------------------------|-----------|--------|---------------|------|--------------|----------------|
| <input type="checkbox"/> | s3-sumit  | None   | Never         | None | None         | Now            |

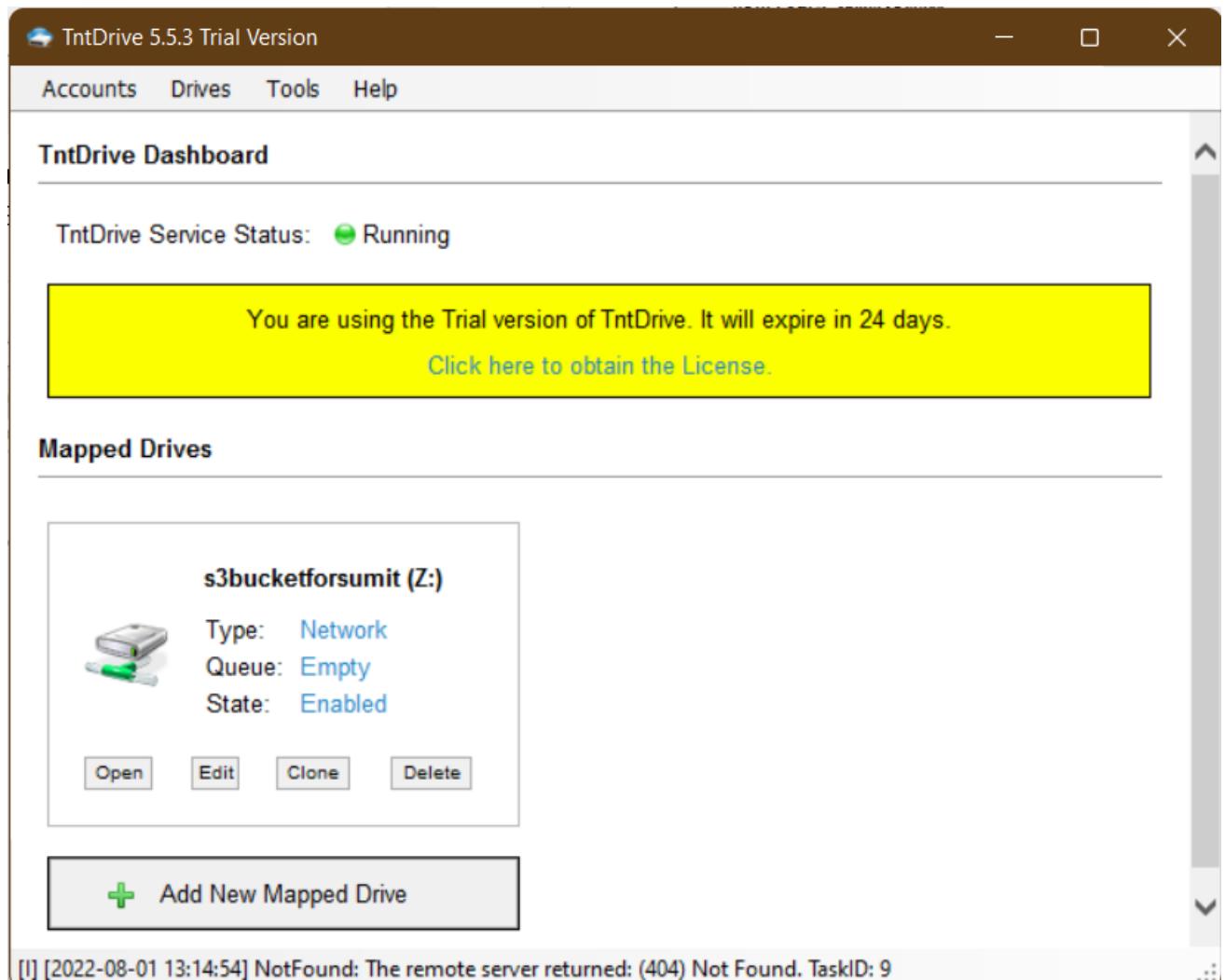
6. Open Tntdrive, select new user, provide the access key and secret access key for the IAM user created and click on add new account.

The screenshot shows the 'Add New Account' interface for Tntdrive. At the top, there's a header bar with a cloud icon, the title 'Add New Account', and standard window controls (minimize, maximize, close). To the right of the title is a link 'online help'. Below the header is a logo consisting of three yellow cubes and a green plus sign. The main area has a heading 'Add New Account' and a sub-instruction 'Enter new account details and click Add new account'. There are several input fields and dropdown menus:

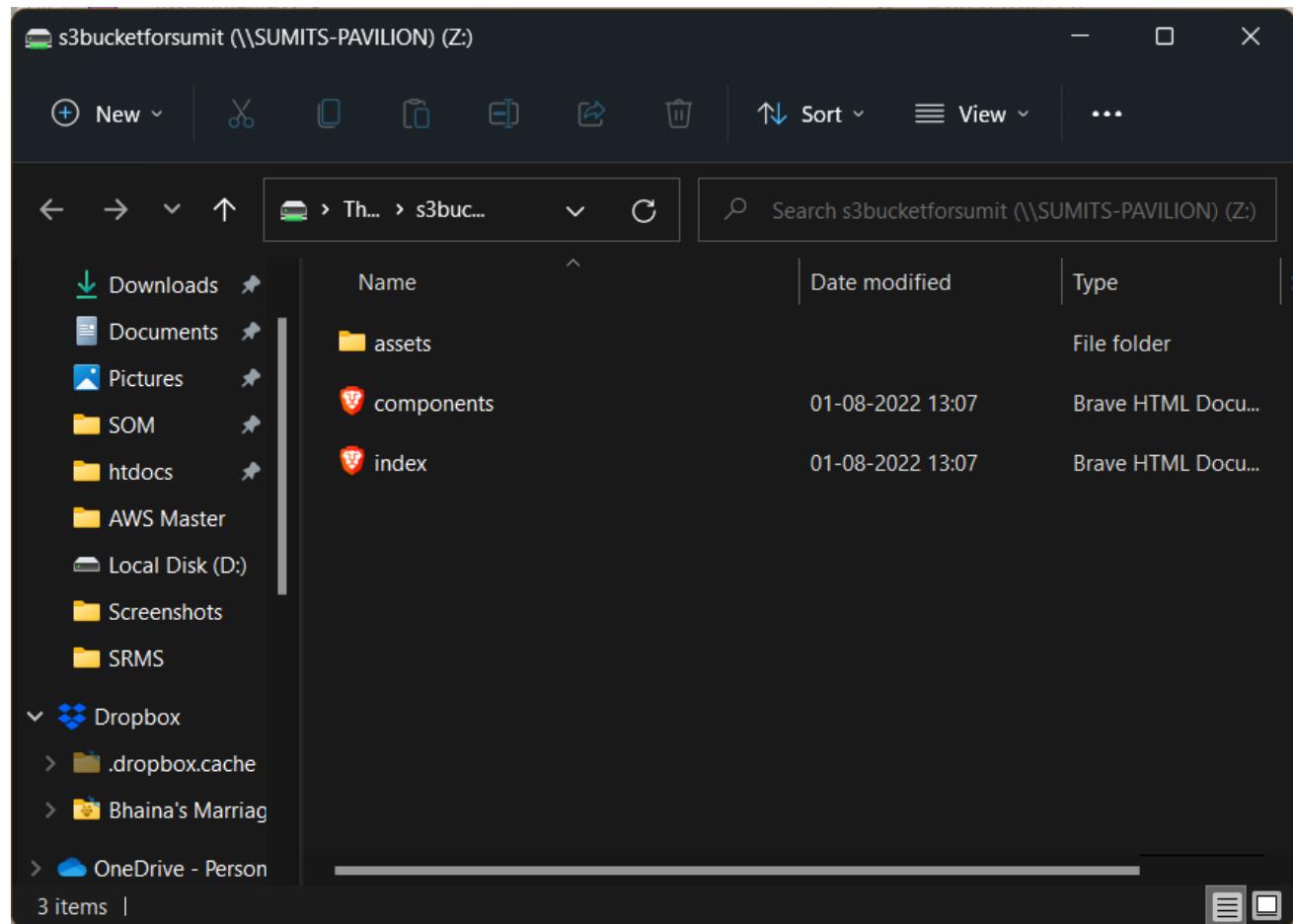
- Account Name:** A text input field containing 'Sumit-s3-iam'. Below it is a placeholder text: 'Assign any name to your account.'
- Account Type:** A dropdown menu set to 'Amazon S3'. Below it is a placeholder text: 'Choose the storage you want to work with. Default is Amazon S3 Storage.'
- Access Key ID:** A text input field containing 'AKIATYFGC35IN4XKI44M'. Below it is a placeholder text: 'Required to sign the requests you send to Amazon S3, see more details at <https://tntdrive.com/keys>'.
- Secret Access Key:** A text input field filled with a long string of black dots ('.....'). Below it is a placeholder text: 'Required to sign the requests you send to Amazon S3, see more details at <https://tntdrive.com/keys>'.
- Use secure transfer (SSL/TLS):** A checked checkbox with the description: 'If checked, all communications with the storage will go through encrypted SSL/TLS channel'.

At the bottom left is a link 'Click here to sign up for Amazon S3'. On the right are two buttons: 'Add new account' with a green checkmark icon and 'Cancel' with a red cancel icon.

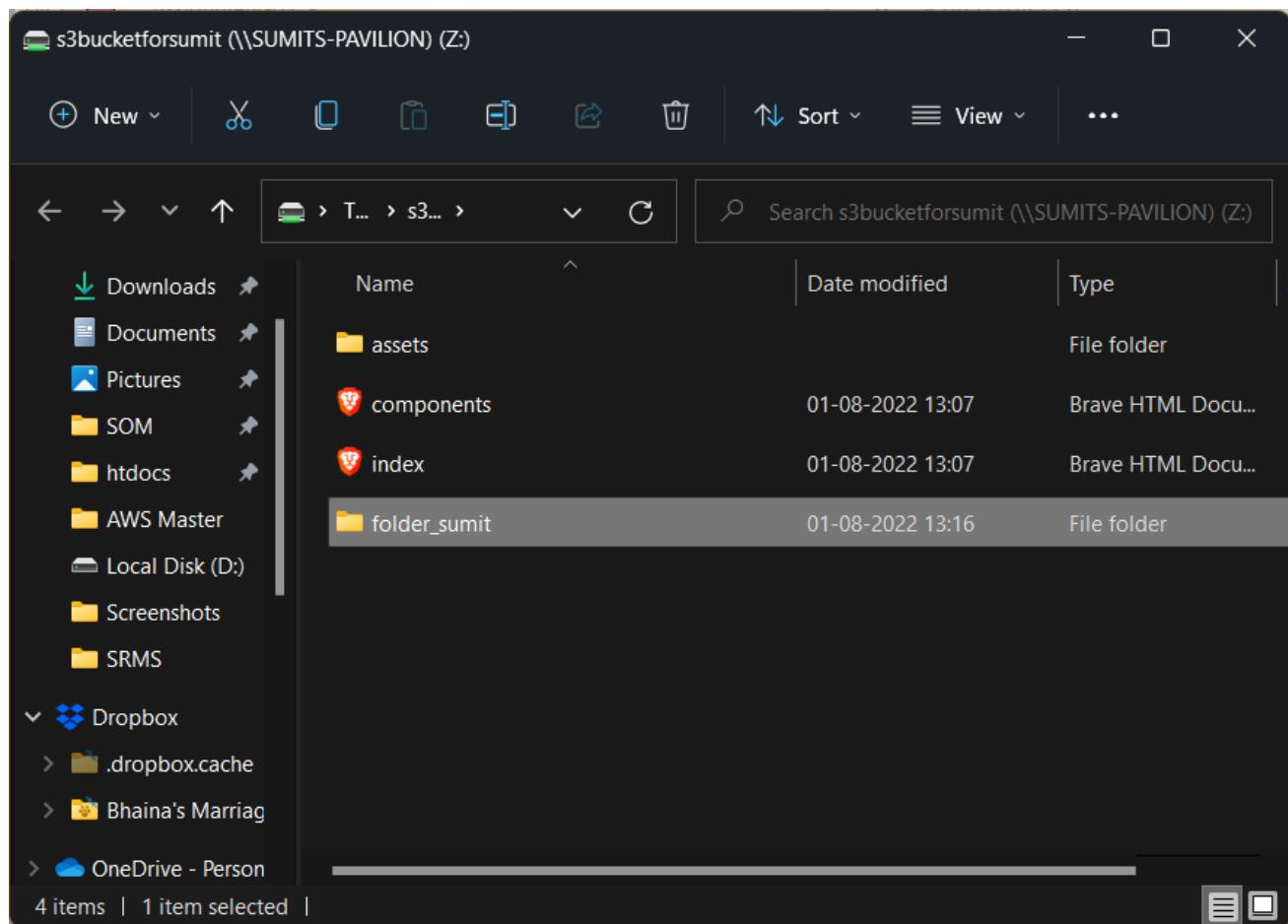
7. Then click on add new mapped drive, type the name of the s3 bucket and click on create.



8. On opening we can see the contents of the bucket on windows file explorer.



9. Adding a folder name `folder_sumit`, we can see that folder being reflected on the AWS S3 bucket console also.



| Name            | Type   | Last modified                        | Size    | Storage class |
|-----------------|--------|--------------------------------------|---------|---------------|
| assets          | Folder | -                                    | -       | -             |
| components.html | html   | August 1, 2022, 13:07:01 (UTC+05:30) | 40.2 KB | Standard      |
| index.html      | html   | August 1, 2022, 13:07:02 (UTC+05:30) | 47.4 KB | Standard      |
| folder_sumit    | Folder | -                                    | -       | -             |