

Name – Majahar Mahamud Kazi

Div – B

Batch – B2

Roll no. – 322036

PRN no. – 22110729

Assignment 4

Aim: Deploy Web application using S3 Bucket.

Theory:

A. What is S3 service of AWS ?

Amazon S3 is a Simple Storage Service in AWS that stores files of different types like Photos, Audio, and Videos as Objects providing more scalability and security to. It allows the users to store and retrieve any amount of data at any point in time from anywhere on the web. It facilitates features such as extremely high availability, security, and simple connection to other AWS Services.

What is Amazon S3 Used for?

Amazon S3 is used for various purposes in the Cloud because of its robust features with scaling and Securing of data. It helps people with all kinds of use cases from fields such as Mobile/Web applications, Big data, Machine Learning and many more. The following are a few Wide Usage of Amazon S3 service.

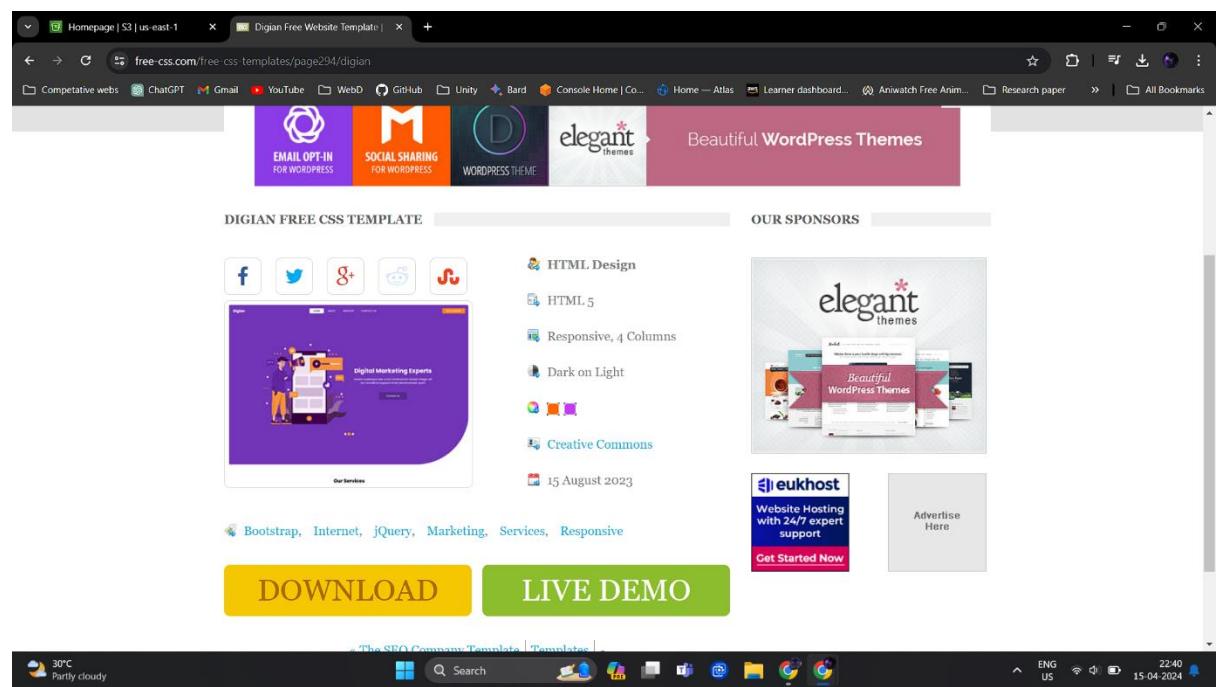
- **Data Storage:** Amazon s3 acts as the best option for scaling both small and large storage applications. It helps in storing and retrieving the data-intensitve applications as per needs in ideal time.



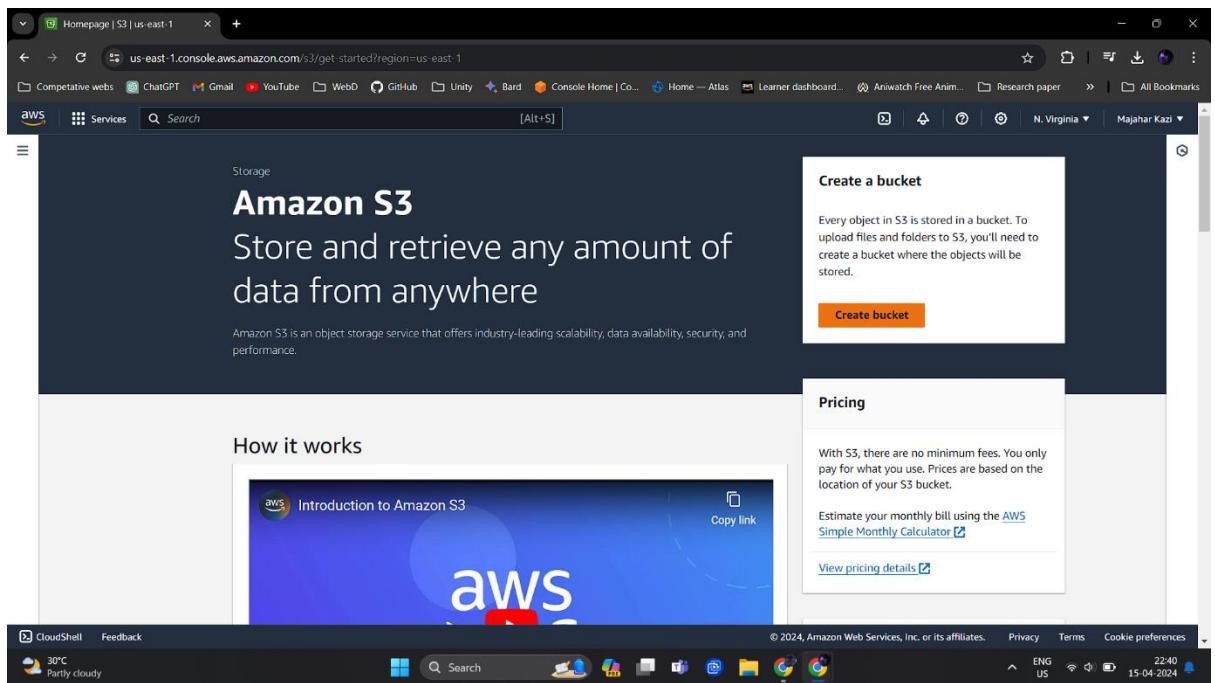
- **Backup and Recovery:** Many Organizations are using Amazon S3 to backup their critical data and maintain the data durability and availability for recovery needs.
- **Hosting Static Websites:** Amazon S3 facilitates in storing HTML, CSS and other web content from Users/developers allowing them for hosting Static Websites benefiting with low-latency access and cost-effectiveness. To know more detailing refer this Article – How to host static websites using.

B. Step-by-step screenshot to upload static web application on the AWS cloud using S3 service.

First we need to download a static website from free-css.com

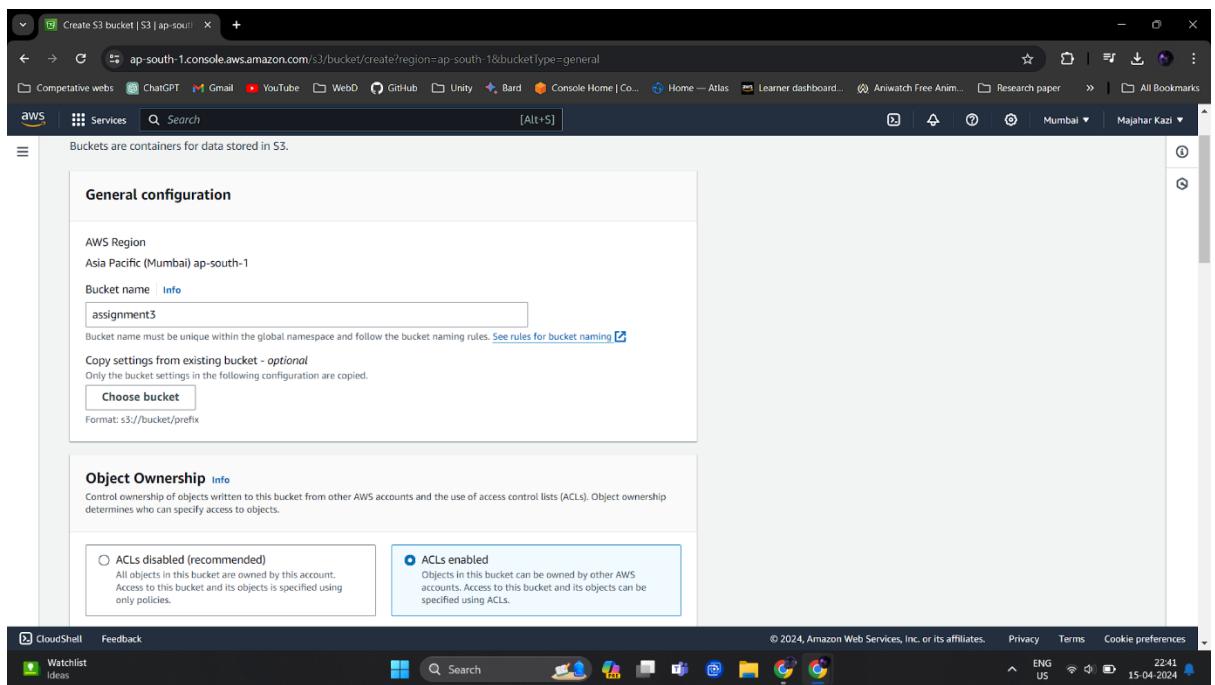


Then we need to create a S3 bucket in aws.



The screenshot shows the Amazon S3 homepage. The main heading is "Amazon S3" with the subtext "Store and retrieve any amount of data from anywhere". Below this, a paragraph explains that Amazon S3 is an object storage service. To the right, there's a "Create a bucket" button and a "Pricing" section. At the bottom, there's a "How it works" section with a video thumbnail titled "Introduction to Amazon S3". The browser status bar at the bottom indicates the URL is "us-east-1.console.aws.amazon.com/s3/get-started?region=us-east-1".

Assign a name to bucket, which should be unique and set the object ownership to acl enabled.



The screenshot shows the "Create S3 bucket" configuration page. Under "General configuration", the "Bucket name" field is set to "assignment". Under "Object Ownership", the "ACLs enabled" option is selected. The browser status bar at the bottom indicates the URL is "ap-south-1.console.aws.amazon.com/s3/bucket/create?region=ap-south-1&bucketType=general".



Remove block all public access and accept the setting.

The screenshot shows the AWS S3 Bucket Creation wizard. In the middle section, titled "Block Public Access settings for this bucket", there is a note: "If you want to enforce object ownership for new objects only, your bucket policy must specify that the bucket-owner-full-control canned ACL is required for object uploads. [Learn more](#)". Below this, there is a section titled "Block all public access" with four options:

- Block public access to buckets and objects granted through new access control lists (ACLs)**: This will block public access permissions applied to newly added buckets or objects, and prevent the creation of new public access ACLs for existing buckets and objects. This setting doesn't change any existing permissions that allow public access to S3 resources using ACLs.
- Block public access to buckets and objects granted through any access control lists (ACLs)**: This will ignore all ACLs that grant public access to buckets and objects.
- Block public access to buckets and objects granted through new public bucket or access point policies**: This will block new bucket and access point policies that grant public access to buckets and objects. This setting doesn't change any existing policies that allow public access to S3 resources.
- Block public and cross-account access to buckets and objects through any public bucket or access point policies**: This will ignore public and cross-account access for buckets or access points with policies that grant public access to buckets and objects.

At the bottom of the page, there is a warning box with an exclamation mark icon: "Turning off block all public access might result in this bucket and the objects within becoming public". It says: "AWS recommends that you turn on block all public access, unless public access is required for specific and verified use cases such as static website hosting." There is a checked checkbox: "I acknowledge that the current settings might result in this bucket and the objects within becoming public." The status bar at the bottom indicates "CloudShell Feedback", "Upcoming Earnings", and the date "15-04-2024".



All the required settings are done for our S3 bucket.

The screenshot shows the 'Create S3 bucket' page in the AWS Management Console. Under 'Default encryption', 'Server-side encryption with Amazon S3 managed keys (SSE-S3)' is selected. Under 'Bucket Key', 'Enable' is selected. A note states that using an S3 Bucket Key for SSE-KMS reduces costs by lowering calls to AWS KMS. A 'Create bucket' button is at the bottom right.

After creating bucket open upload section.

The screenshot shows the 'Objects' tab of the 'assignment3majahar' bucket. It displays a table with columns for Name, Type, Last modified, Size, and Storage class. A large orange 'Upload' button is prominently displayed at the bottom center. The status bar at the bottom indicates 'CloudShell' and 'Feedback'.



Go to add files.

Add the files and folders you want to upload to S3. To upload a file larger than 160GB, use the AWS CLI, AWS SDK or Amazon S3 REST API. [Learn more](#)

Drag and drop files and folders you want to upload here, or choose [Add files](#) or [Add folder](#).

Files and folders (0)

All files and folders in this table will be uploaded.

| Name | Folder | Type |
|---------------------|--------|------|
| No files or folders | | |

You have not chosen any files or folders to upload.

Destination

s3://assignment3majahar

Now upload the downloaded website's zip file.

Upload succeeded

The information below will no longer be available after you navigate away from this page.

Summary

| Destination | Succeeded | Failed |
|-------------------------|--------------------------|-------------------|
| s3://assignment3majahar | 1 file, 1.2 MB (100.00%) | 0 files, 0 B (0%) |

Files and folders (1 Total, 1.2 MB)

| Name | Folder | Type | Size | Status | Error |
|---------|--------|-----------------|--------|-----------|-------|
| web.zip | - | application/... | 1.2 MB | Succeeded | - |



From actions do to make public acl section for our web.zip file

The screenshot shows the AWS S3 console interface. On the left, there's a sidebar with various options like 'Access Grants', 'Storage Lens', and 'Feature spotlight'. The main area displays the 'assignment3majahar' bucket with one object named 'web.zip'. A context menu is open over this object, with the 'Actions' dropdown expanded. Within the 'Actions' dropdown, the 'Make public using ACL' option is highlighted with a blue box.

The screenshot shows the 'Make public' dialog box from the AWS S3 console. It contains a warning message about enabling public access and a table of specified objects. The 'web.zip' file is listed in the table. At the bottom right of the dialog, the 'Make public' button is highlighted with a blue box.



Create an EC2 instance and assign a name to it.

The screenshot shows two browser windows for the AWS EC2 console. The top window displays the main EC2 dashboard with resource counts and a 'Launch instance' button. The bottom window shows the 'Launch an instance' configuration page, where the instance name is set to 'assignment3'. The configuration includes an Amazon Linux 2023 AMI, t2.micro virtual server type, and a new security group. A tooltip indicates a free tier benefit for the first year. The status bar at the bottom shows the date as 15-04-2024.

EC2 Dashboard

- EC2 Global View
- Events
- Instances**
 - Instances
 - Instance Types
 - Launch Templates
 - Spot Requests
 - Savings Plans
 - Reserved Instances
 - Dedicated Hosts
 - Capacity Reservations **New**
- Images**
 - AMIs
 - AMI Catalog
- Elastic Block Store**
 - Volumes
 - Snapshots

Resources

You are using the following Amazon EC2 resources in the Asia Pacific (Mumbai) Region:

| | | | | | |
|---------------------|---|---------------------|---|-----------------|---|
| Instances (running) | 0 | Auto Scaling Groups | 0 | Dedicated Hosts | 0 |
| Elastic IPs | 0 | Instances | 0 | Key pairs | 4 |
| Load balancers | 0 | Placement groups | 0 | Security groups | 0 |
| Snapshots | 0 | Volumes | 0 | | |

Launch instance

To get started, launch an Amazon EC2 Instance, which is a virtual server in the cloud.

Service health

AWS Health Dashboard

Region
Asia Pacific (Mumbai)

Status
This service is operating normally.

Zones

EC2 Free Tier Info
Offers for all AWS Regions.

2 EC2 free tier offers in use

End of month forecast
0 offers forecasted to exceed free tier limit.

Exceeds free tier
0 offers exceeded and is now pay-as-you-go pricing.

[View Global EC2 resources](#)

Offer usage (monthly)

Linux EC2 Instances 0%
748,456,389 hours remaining

Storage space on EBS 0%
29.98 GB remaining

[View all AWS Free Tier offers](#)

Launch an instance | EC2 | ap-south-1 | Make objects public - S3 bucket

Launch an instance

Amazon EC2 allows you to create virtual machines, or instances, that run on the AWS Cloud. Quickly get started by following the simple steps below.

Name and tags Info

Name
assignment3 [Add additional tags](#)

Application and OS Images (Amazon Machine Image) Info

An AMI is a template that contains the software configuration (operating system, application server, and applications) required to launch your instance. Search or Browse for AMIs if you don't see what you are looking for below.

Search our full catalog including 1000s of application and OS images

Quick Start

Summary

Number of instances Info
1

Software Image (AMI)
Amazon Linux 2023 AMI 2023.4.2... [read more](#)
ami-09298640a92b2d12c

Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

Storage (volumes)
1 volume(s) - 8 GiB

Free tier: In your first year X
includes 750 hours of t2.micro (or t4.micro in the Regions in which

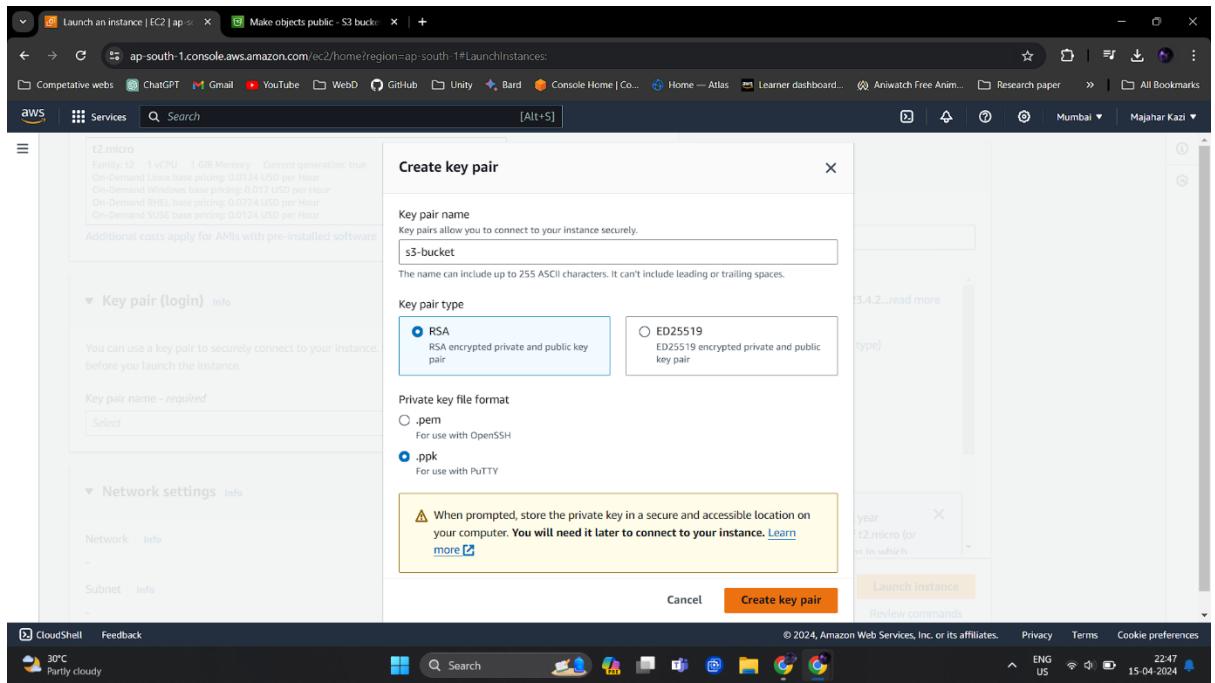
Launch instance [Review commands](#)

© 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

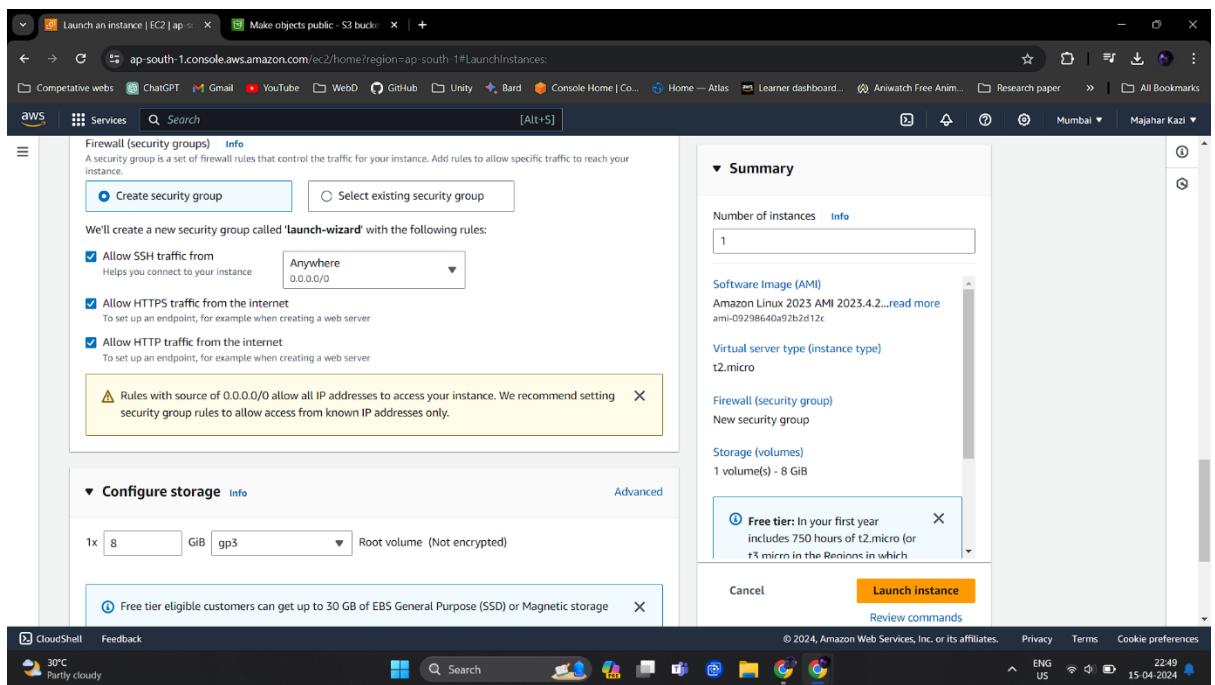
CloudShell Feedback 30°C Partly cloudy ENG US 22:47 15-04-2024



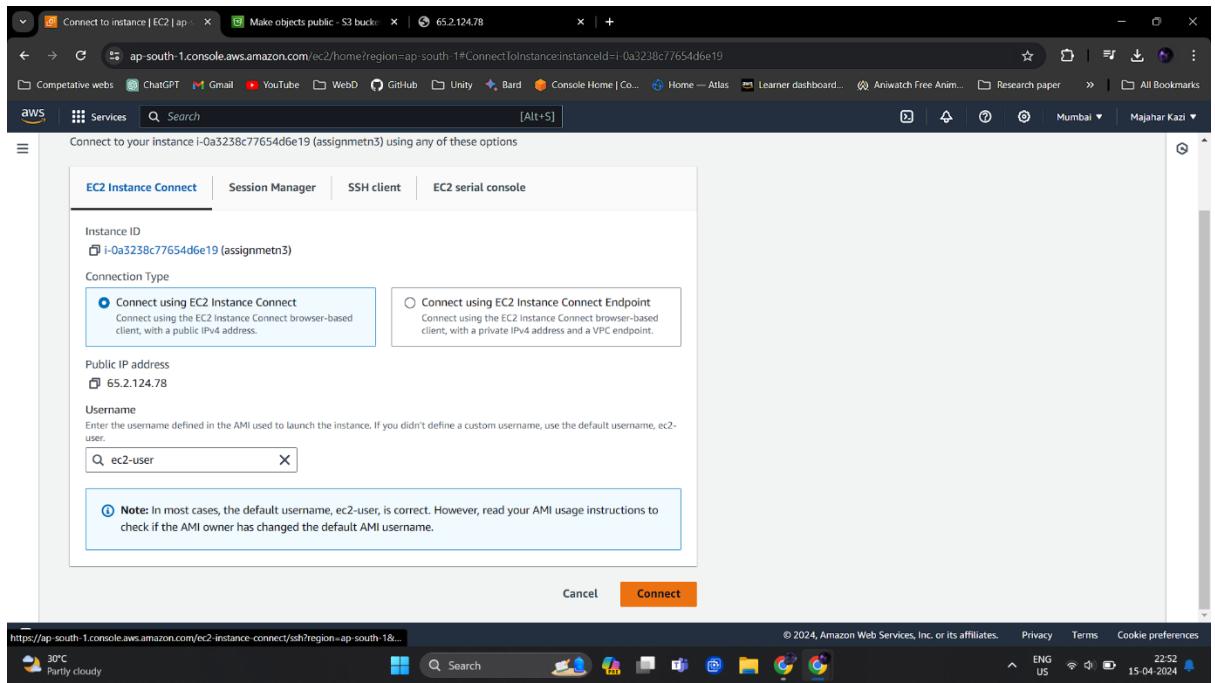
Create a key with .ppk extension



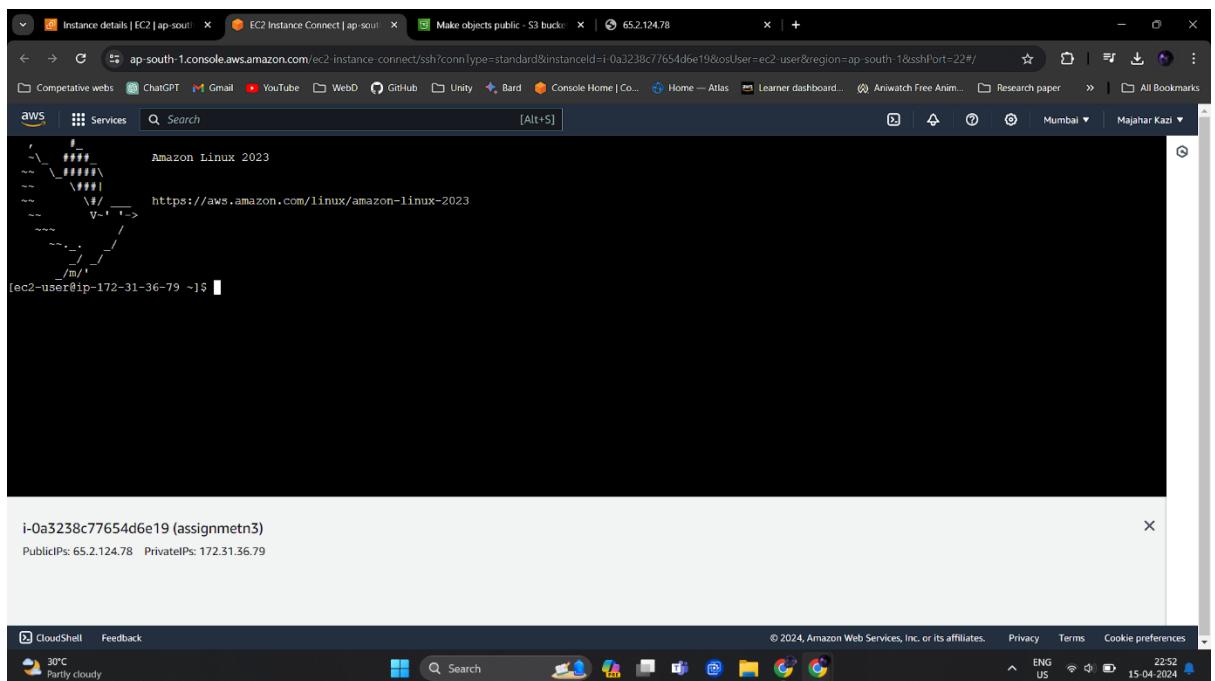
In network settings allow traffic from http and https and launch the instance.



Now connect to our created Virtual Machine(VM).



We are into our vm now.



Let's initialize the vm by updating system and installing httpd.

```
[ec2-user@ip-172-31-36-79 ~]$ sudo su
[root@ip-172-31-36-79 ec2-user]# yum update -y
Last metadata expiration check: 0:01:57 ago on Mon Apr 15 17:21:04 2024.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-36-79 ec2-user]# yum install -y httpd
Last metadata expiration check: 0:02:34 ago on Mon Apr 15 17:21:04 2024.
Dependencies resolved.

Package          Architecture      Version           Repository      Size
Installing:
httpd            x86_64          2.4.58-1.amzn2023   amazonlinux    47 k
Installing dependencies:

```

i-0a3238c77654d6e19 (assignmetn3)
PublicIPs: 65.2.124.78 PrivateIPs: 172.31.36.79

Let's download our web.zip file from our S3 bucket. For that we need to copy the url from S3 bucket for our web.zip file.

Amazon S3 > Buckets > assignment3majahar

Objects (1) Info

| Name | Type | Last modified | Size | Storage class |
|---------|------|-----------------------------------------|--------|---------------|
| web.zip | zip | April 15, 2024, 22:44:47 (UTC+05:30) | 1.2 MB | Standard |



```

Verifying : mailcap-2.1.49-3.amzn2023.0.3.noarch
Verifying : mod_http2-2.0.11-2.amzn2023.x86_64
Verifying : mod_lua-2.4.58-1.amzn2023.x86_64

Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64
  generic-logos-httd-18.0.1-2.amzn2023.0.3.noarch
  httpd-filesystem-2.4.58-1.amzn2023.noarch
  mailcap-2.1.49-3.amzn2023.0.3.noarch
  apr-util-1.6.3-1.amzn2023.0.1.x86_64
  httpd-2.4.58-1.amzn2023.x86_64
  httpd-tools-2.4.58-1.amzn2023.x86_64
  mod_http2-2.0.11-2.amzn2023.x86_64
  apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64
  httpd-core-2.4.58-1.amzn2023.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64
  mod_lua-2.4.58-1.amzn2023.x86_64

Complete!
[root@ip-172-31-36-79 ec2-user]# cd /var/www/html
[root@ip-172-31-36-79 html]# wget https://assignment3majahar.s3.ap-south-1.amazonaws.com/web.zip
--2024-04-15 17:27:00-- https://assignment3majahar.s3.ap-south-1.amazonaws.com/web.zip
Resolving assignment3majahar.s3.ap-south-1.amazonaws.com (assignment3majahar.s3.ap-south-1.amazonaws.com)... 16.12.40.26, 16.12.40.90, 16.12.40.110, ...
Connecting to assignment3majahar.s3.ap-south-1.amazonaws.com (assignment3majahar.s3.ap-south-1.amazonaws.com) |16.12.40.26|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1288261 (1.2M) [application/zip]
Saving to: 'web.zip'

web.zip          100%[=====]  1.23M --.-KB/s   in 0.01s

2024-04-15 17:27:01 (82.5 MB/s) - 'web.zip' saved [1288261/1288261]

[root@ip-172-31-36-79 html]#

```

i-0a3238c77654d6e19 (assignmetn3)
PublicIPs: 65.2.124.78 PrivateIPs: 172.31.36.79

We got our web's zip file lets unzip it.

```

[root@ip-172-31-36-79 html]# unzip web.zip
Archive: web.zip
  creating: digian-html/
  inflating: digian-html/about.html
  inflating: digian-html/contact.html
  creating: digian-html/css/
  inflating: digian-html/css/bootstrap.css
  inflating: digian-html/css/font-awesome.min.css
  inflating: digian-html/css/responsive.css
  inflating: digian-html/css/style.css
  inflating: digian-html/css/style.css.map
  inflating: digian-html/css/style.scss
  creating: digian-html/fonts/
  inflating: digian-html/fonts/fontawesome-webfont.ttf
  inflating: digian-html/fonts/fontawesome-webfont.woff
  inflating: digian-html/fonts/fontawesome-webfont.woff2
  creating: digian-html/images/
  inflating: digian-html/images/about-img.jpg
  inflating: digian-html/images/about-img.png
  inflating: digian-html/images/case-1.jpg
  inflating: digian-html/images/case-2.jpg
  inflating: digian-html/images/client.jpg
  inflating: digian-html/images/contact-img.png
  inflating: digian-html/images/s1.png

i-0a3238c77654d6e19 (assignmetn3)
PublicIPs: 65.2.124.78 PrivateIPs: 172.31.36.79

```

i-0a3238c77654d6e19 (assignmetn3)
PublicIPs: 65.2.124.78 PrivateIPs: 172.31.36.79



Let's copy all the files from unzipped folder to var/www/http folder and remove both zip file and unzipped directory.

```
inflating: digian-html/Fonts/Fontawesome-webFont.woff2
creating: digian-html/images/
inflating: digian-html/images/about-img.jpg
inflating: digian-html/images/about-img.png
inflating: digian-html/images/case-1.jpg
inflating: digian-html/images/case-2.jpg
inflating: digian-html/images/client.jpg
inflating: digian-html/images/contact-img.png
inflating: digian-html/images/s1.png
inflating: digian-html/images/s2.png
inflating: digian-html/images/s3.png
inflating: digian-html/images/s4.png
inflating: digian-html/images/slider-bg.jpg
inflating: digian-html/images/slider-img.png
inflating: digian-html/index.html
creating: digian-html/js/
inflating: digian-html/js/bootstrap.js
inflating: digian-html/js/custom.js
inflating: digian-html/js/jquery-3.4.1.min.js
inflating: digian-html/service.html
[root@ip-172-31-36-79 html]# ls
digian-html web.zip
[root@ip-172-31-36-79 html]# cp -r digian-html/* /var/www/html/
[root@ip-172-31-36-79 html]# rm -rf digian-html web.zip
[root@ip-172-31-36-79 html]#
```

i-0a3238c77654d6e19 (assignmetn3)
PublicIPs: 65.2.124.78 PrivateIPs: 172.31.36.79



In end enable httpd for launching our web.

```
Docs: man:httdp.service(8)
[root@ip-172-31-36-79 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-36-79 html]# systemctl start httpd
[root@ip-172-31-36-79 html]# systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Mon 2024-04-15 17:31:37 UTC; 5s ago
     Docs: man:httdp.service(8)
Main PID: 26237 (httpd)
      Status: "Started, listening on: port 80"
     Tasks: 177 (limit: 1114)
    Memory: 12.9M
      CPU: 63ms
     CGroup: /system.slice/httpd.service
             ├─26237 /usr/sbin/httpd -DFOREGROUND
             ├─26238 /usr/sbin/httpd -DFOREGROUND
             ├─26239 /usr/sbin/httpd -DFOREGROUND
             ├─26240 /usr/sbin/httpd -DFOREGROUND
             ├─26241 /usr/sbin/httpd -DFOREGROUND

Apr 15 17:31:37 ip-172-31-36-79.ap-south-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Apr 15 17:31:37 ip-172-31-36-79.ap-south-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Apr 15 17:31:37 ip-172-31-36-79.ap-south-1.compute.internal httpd[26237]: Server configured, listening on: port 80
[root@ip-172-31-36-79 html]#
```

i-0a3238c77654d6e19 (assignmetn3)
PublicIPs: 65.2.124.78 PrivateIPs: 172.31.36.79



Copy the Auto-assigned IP address for EC2 instance.

The screenshot shows the AWS EC2 Instances details page for an instance named 'i-0a3238c77654d6e19 (assignmetn3)'. The instance is running and has an auto-assigned public IPv4 address of 65.2.124.78. Other details include a private IP of 172.31.36.79, a private DNS name of ip-172-31-36-79.ap-south-1.compute.internal, and a VPC ID of vpc-093afe7dc5b88cb1c. The instance type is t2.micro. The interface also shows the AWS Compute Optimizer finding status and a link to learn more.

Result:



Instance details | EC2 | ap-sou... | EC2 Instance Connect | ap-sou... | assignment3majshar - S3 buck... | Digian

Not secure 65.2124.78

Competitive webs ChatGPT Gmail YouTube WebD GitHub Unity Bard Console Home | Co... Home — Atlas Learner dashboard... Aniwatch Free Anim... Research paper All Bookmarks

Digian

HOME ABOUT SERVICES CONTACT US GET A QUOTE

Digital Marketing Experts

Aenean scelerisque felis ut orci condimentum laoreet. Integer nisi, convallis et augue sit amet, lobortis semper quam.

Contact Us

Our Services

consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Link Building
Lorem ipsum dolor sit amet, consectetur adipiscing elit,
[Read More →](#)

On page SEO
Lorem ipsum dolor sit amet, consectetur adipiscing elit,
[Read More →](#)

Online Marketing
Lorem ipsum dolor sit amet, consectetur adipiscing elit,
[Read More →](#)

Email Marketing
Lorem ipsum dolor sit amet, consectetur adipiscing elit,
[Read More →](#)

[View More](#)

About Us

There are many variations of passages of Lorem Ipsum available, but the majority have suffered alteration in some form, by injected humour, or randomised words which don't look even slightly believable. If you are going to use a passage of Lorem Ipsum, you need to be sure there isn't anything embarrassing hidden in the middle of text. All

[Read More](#)

29°C Partly cloudy

Search

ENG US 23:02 15-04-2024



Our Case Studies



Sit amet consectetur adipisciing elit

Alteration in some form, by injected humour, or randomised words which don't look even slightly believable.

[Read More →](#)



Excepturi placeat nihil eos maxime

Alteration in some form, by injected humour, or randomised words which don't look even slightly believable.

[Read More →](#)

Testimonial



MOROJINK

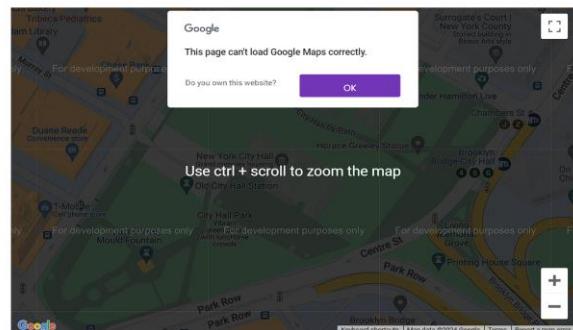
“

Lorem ipsum dolor sit amet, consectetur adipisciing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepturi placeat nihil eos maxime.

• • •

Request A Call back

[SEND](#)



Digian

Necessary, making this the first true generator on the Internet. It uses a dictionary of over 200 Latin words, combined with a handful

Links

[Home](#)
[About](#)
[Services](#)
[Contact Us](#)

Subscribe

Enter email

[Subscribe](#)

Address

Location
Call +01 1234567890
demo@gmail.com
[f](#) [t](#) [in](#) [r](#)

© 2024 All Rights Reserved By Free Html Templates



Author – Majahar Kazi

MONDAY 15 April 2024 18:04:08 PM IST

