SIBA CAMPUS

SRI LANKA INTERNATIONAL BUDDHIST ACADEMY
(Higher Educational Institute of Sri Dalada Maligawa)
Pallekele, Kundasale, Sri Lanka

END SEMESTER EVALUATION 2024 (II)

IT 246 - CLOUD COMPUTING

SUBMITTED BY

D.A.I.M.DASANAYAKE

(BSC/WD/22/36/13)

To the

DEPARTMENT OF INFORMATION TECHNOLOGY

In partial fulfillment of the requirement for
The award of the degree of

BSc degree in Information Technology

Of the

SRI LANKA INTERNATIONAL BUDDHIST ACADEMY

(SIBA CAMPUS)

PALLEKELE

SRI LANKA

2024

Contents

FIGURES LIST

ABSTRACT

This case study delves into establishing a personal blog using Amazon Web Services (AWS), employing an EC2 instance for hosting and an S3 bucket for storing and delivering static content. It illustrates the setup process, including IAM role creation, EC2 instance launch and setup, and Apache web server installation. Furthermore, it outlines the creation and setup of an S3 bucket for streamlined file storage. This guide equips users with fundamental web development and server management skills to deploy a functional personal blog, capitalizing on AWS's scalability, reliability, and seamless integration capabilities. Ultimately, it offers a robust and economical solution for sharing personal content.

INTRODUCTION

This case study walks you through creating a personal blog on AWS, offering a dependable and scalable method for sharing your ideas and stories online. It covers setting up a blog hosted on an EC2 instance, with images and static files served from an S3 bucket. By following these instructions, you'll configure your AWS environment, install and configure a web server, set up an S3 bucket, and deploy a basic HTML website. Geared towards individuals with basic web development and server management skills, this project provides practical exposure to AWS's robust and effective infrastructure.

SET UP THE AWS ENVIRONMENT

Establishing the AWS environment marks the initial phase in hosting a personal blog on an EC2 instance while serving static files from an S3 bucket. This involves creating an IAM role and launching an EC2 instance.

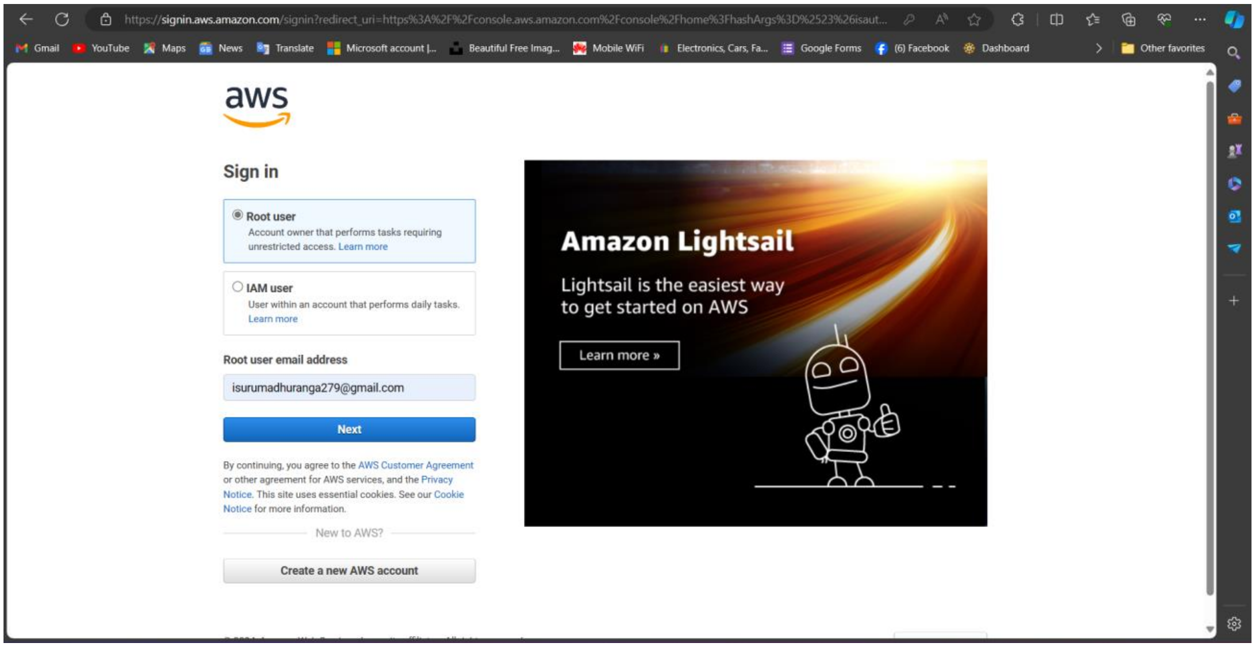
IAM assumes a pivotal role in securely managing access to AWS services and resources. Crafting an IAM role tailored for EC2 instances ensures the instance possesses the requisite permissions to interact with other AWS services, notably S3. This role facilitates EC2 instance access to the S3 bucket housing static files like images.

The launch of an EC2 instance forms the bedrock for hosting the web server responsible for serving the personal blog. Users navigate the AWS Management Console to select the desired Amazon Machine Image (AMI) and instance type, configure security settings, and specify additional details, including IAM role assignments. Configuring security groups is vital to permit inbound traffic on port 80 (HTTP) for web traffic and port 22 (SSH) for remote access.

By adeptly setting up the AWS environment, users establish a secure and scalable groundwork for hosting their personal blog, setting the stage for subsequent steps like web server installation, configuration, and S3 bucket setup.

01. SET UP THE AWS ENVIRONMENT

- ## Create AWS Environment



Sign in

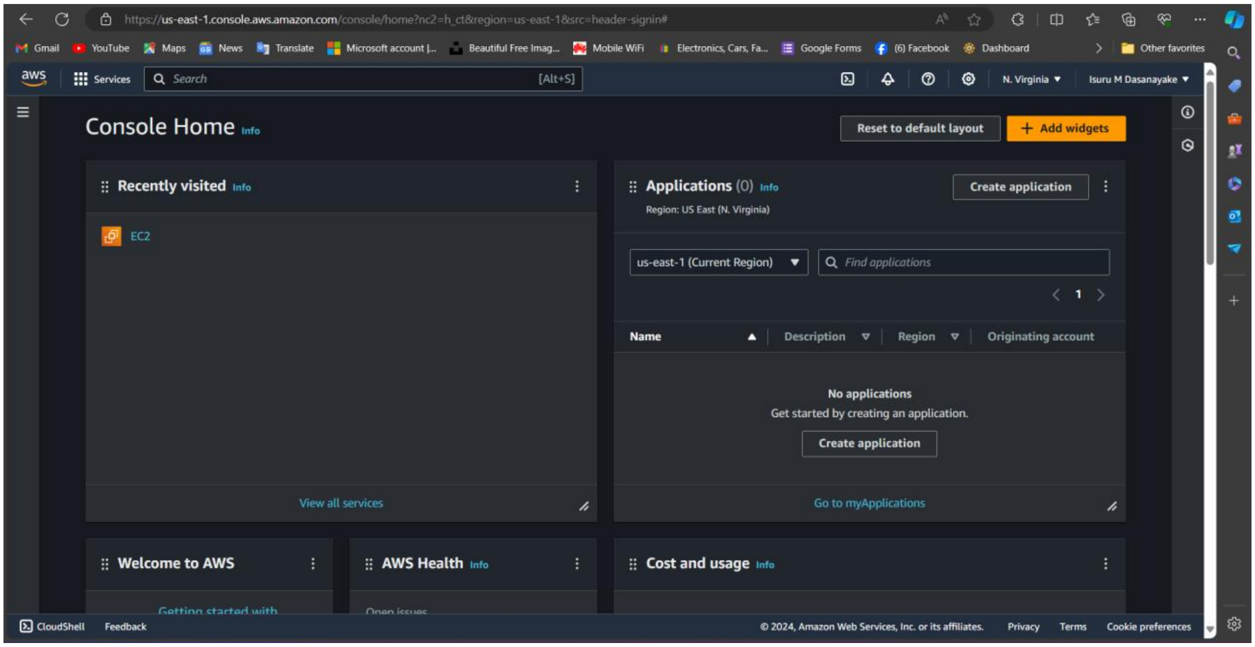
● Root user
Account owner that performs tasks requiring unrestricted access. Learn more

○ IAM user
User within an account that performs daily tasks. Learn more

Root user email address: isurumadhuranga279@gmail.com

By continuing, you agree to the AWS Customer Agreement or other agreement for AWS services, and the Privacy Notice. This site uses essential cookies. See our Cookie Notice for more information.

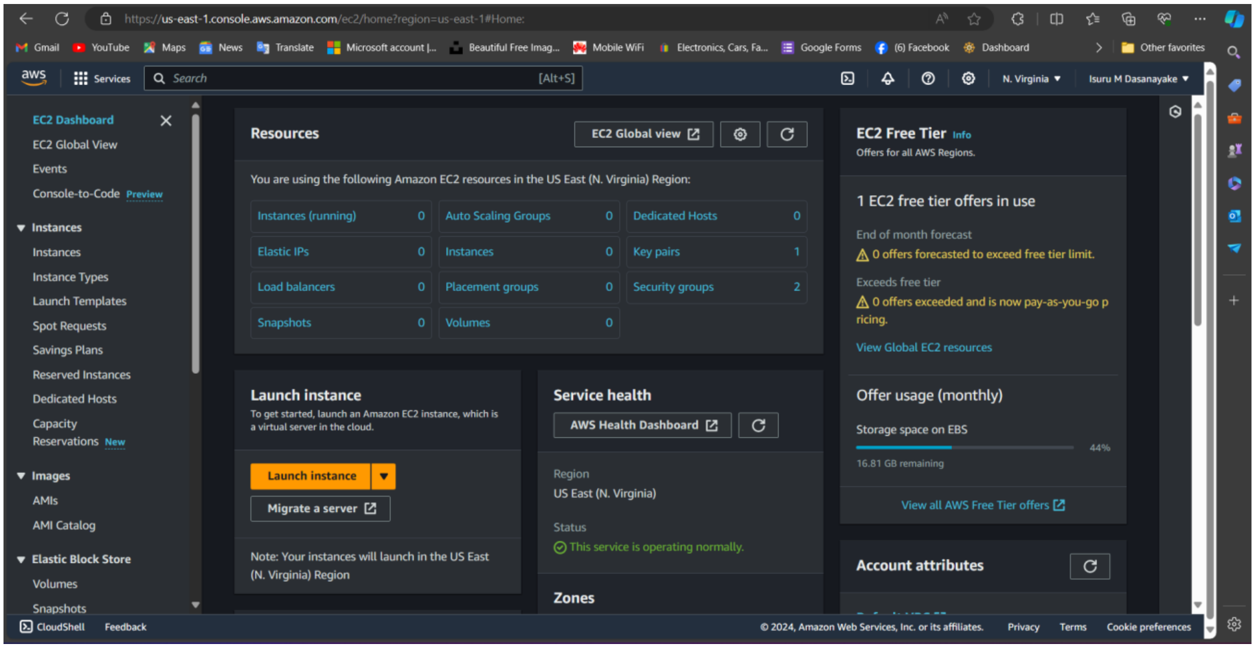
1.0 Sign In to AWS



No applications
Get started by creating an application.

1.01 Console Home

- **Launch an EC2 instance**



Resources

You are using the following Amazon EC2 resources in the US East (N. Virginia) Region:

Instances (running)	0	Auto Scaling Groups	0	Dedicated Hosts	0
Elastic IPs	0	Instances	0	Key pairs	1
Load balancers	0	Placement groups	0	Security groups	2
Snapshots	0	Volumes	0		

Launch instance

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

Note: Your instances will launch in the US East (N. Virginia) Region

Service health

Region
US East (N. Virginia)

Status
This service is operating normally.

EC2 Free Tier Info
Offers for all AWS Regions.

1 EC2 free tier offers in use

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

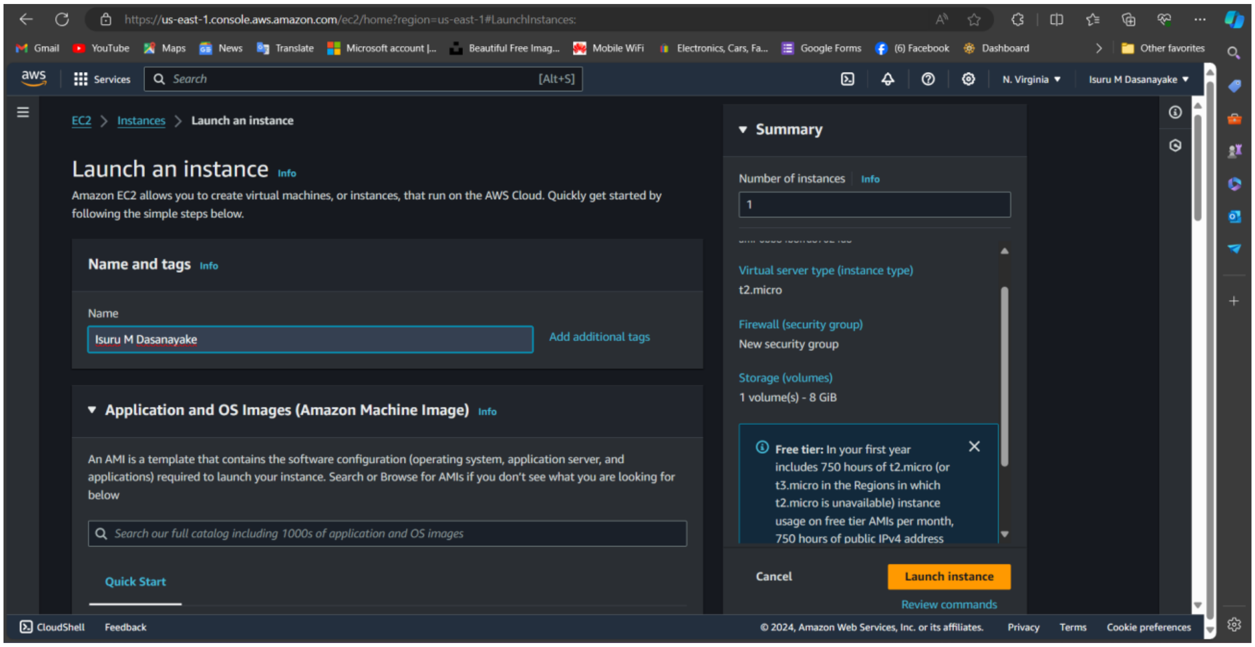
Exceeds free tier
⚠ 0 offers exceeded and is now pay-as-you-go pricing.

Offer usage (monthly)

Storage space on EBS 44%
16.81 GB remaining

Account attributes

1.02 Figures EC2 Dashboard



Launch an instance Info

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: Isuru M Dasanayake

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

Summary

Number of instances Info: 1

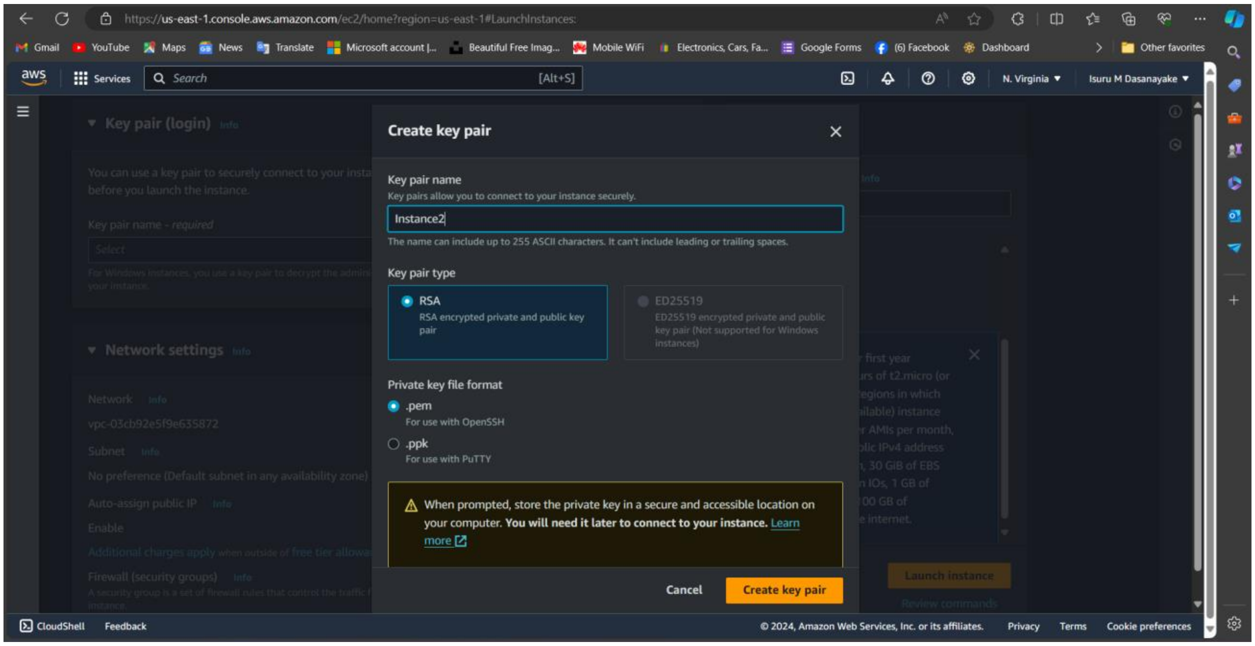
Virtual server type (instance type)
t2.micro

Firewall (security group)
New security group

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

Free tier: In your first year includes 750 hours of t2.micro (or t3.micro in the Regions in which t2.micro is unavailable) instance usage on free tier AMIs per month, 750 hours of public IPv4 address

1.03 Figures launch an instance


https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Services
Search
[Alt+S]
N. Virginia
Isuru M Dasanayake
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Create key pair

Key pair name

Key pairs allow you to connect to your instance securely.

Instance2

The name can include up to 255 ASCII characters. It can't include leading or trailing spaces.

Key pair type

● RSA
RSA encrypted private and public key pair

○ ED25519
ED25519 encrypted private and public key pair (Not supported for Windows instances)

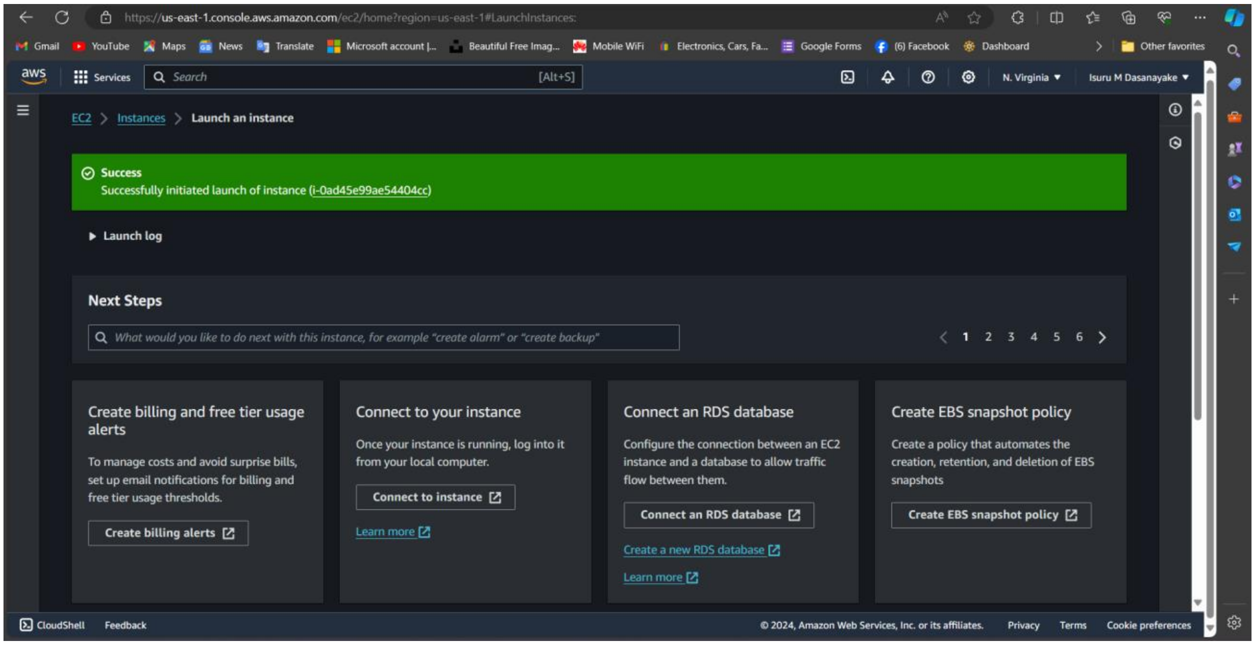
Private key file format

● .pem
For use with OpenSSH

○ .ppk
For use with PuTTY

⚠ When prompted, store the private key in a secure and accessible location on your computer. **You will need it later to connect to your instance.** Learn more

1.04 Figures Create Key pair


https://us-east-1.console.aws.amazon.com/ec2/home?region=us-east-1#LaunchInstances:
Services
Search
[Alt+S]
N. Virginia
Isuru M Dasanayake
EC2 > Instances > Launch an instance
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Success
Successfully initiated launch of instance (i-0ad45e99ae54404cc)

▶ Launch log

Next Steps

What would you like to do next with this instance, for example "create alarm" or "create backup"

Create billing and free tier usage alerts

To manage costs and avoid surprise bills, set up email notifications for billing and free tier usage thresholds.

Connect to your instance

Once your instance is running, log into it from your local computer.

Connect an RDS database

Configure the connection between an EC2 instance and a database to allow traffic flow between them.

Create EBS snapshot policy

Create a policy that automates the creation, retention, and deletion of EBS snapshots

1.05 Figures Successfully Launch Instance

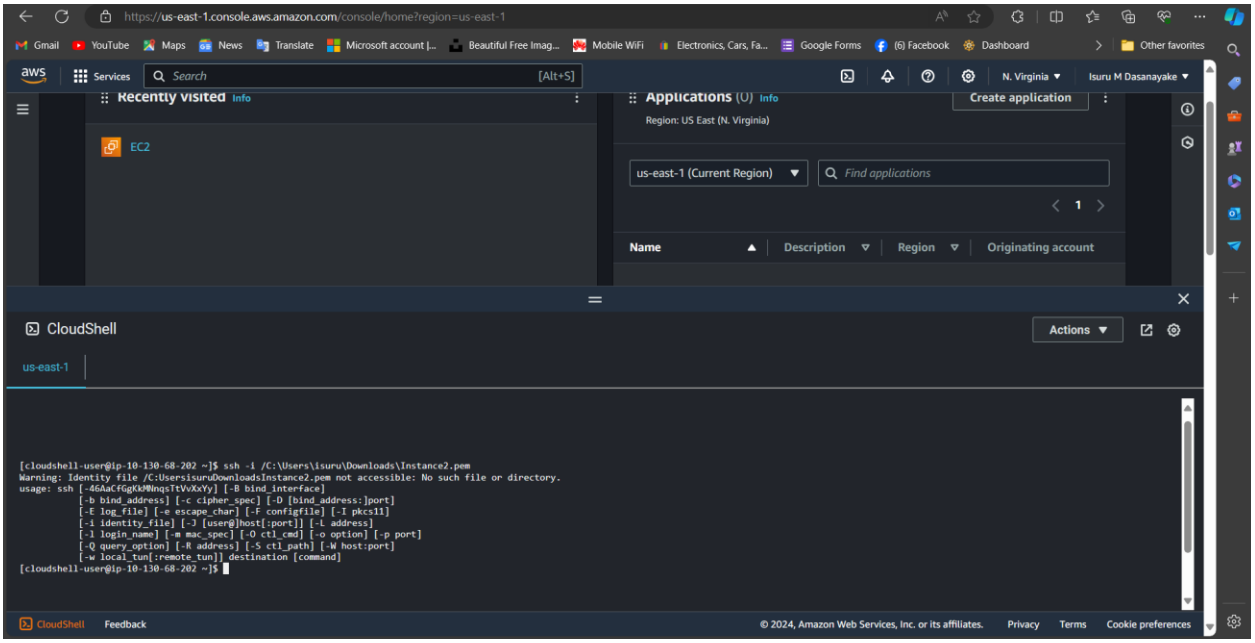
INSTALL A WEB SERVER- APACHE

Installing a web server is a vital stage in configuring your personal blog on an AWS EC2 instance, and for this project, Apache, a widely used and dependable web server, will be employed. Upon launching and connecting to your EC2 instance via SSH, the initial task involves updating the package lists to ensure all software is current, accomplished with the command ‘sudo yum update –y’. Subsequently, installing Apache is a straightforward process with ‘sudo yum install httpd –y’. Once installed, initiating the web server is achieved by starting the Apache service with ‘sudo systemctl start httpd’, enabling it to serve web pages. To ensure Apache automatically starts on boot, `sudo systemctl enable httpd` is executed.

Validating the Apache installation is crucial to confirm the correct setup and operation of the web server. This entails accessing the public IP address of the EC2 instance in a web browser. If Apache is correctly installed and operational, the default Apache test page will be displayed, indicating readiness to host the blog. This stage establishes the core capability of the EC2 instance to deliver web content, paving the way for deploying the HTML website and integrating with the S3 bucket for static file storage.

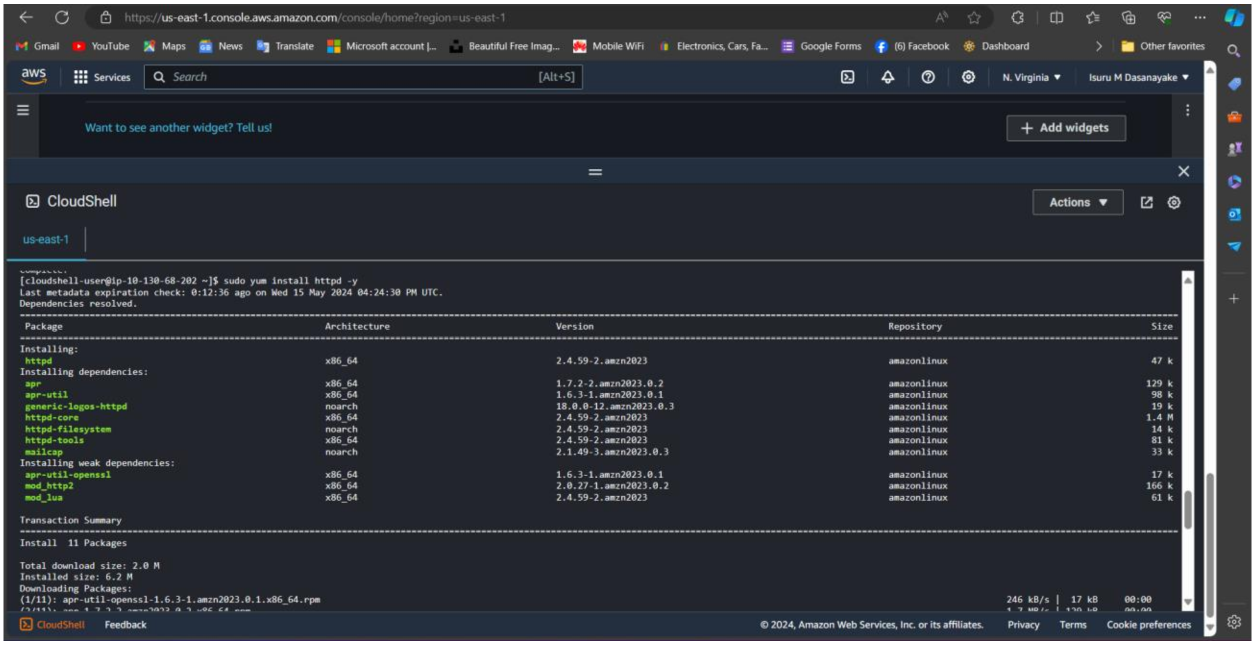
INSTALL A WEB SERVER- APACHE

- **Connect to EC2 instance via SSH**



```
[cloudshell-user@ip-10-130-68-202 ~]$ ssh -i /C:\Users\isuru\Downloads\Instance2.pem
Warning: Identity file /C:UsersisuruDownloadsInstance2.pem not accessible: No such file or directory.
usage: ssh [-46AaCfGgKkMNnqsTtVvXxYy] [-B bind_interface]
           [-b bind_address] [-c cipher_spec] [-D [bind_address:]port]
           [-E log_file] [-e escape_char] [-F configfile] [-I pkcs11]
           [-i identity_file] [-J [user@]host[:port]] [-L address]
           [-l login_name] [-m mac_spec] [-O ctl_cmd] [-o option] [-p port]
           [-Q query_option] [-R address] [-S ctl_path] [-W host:port]
           [-w local_tun[:remote_tun]] destination [command]
[cloudshell-user@ip-10-130-68-202 ~]$ 
```

1.06 Figures CloudShell Amazon Linux

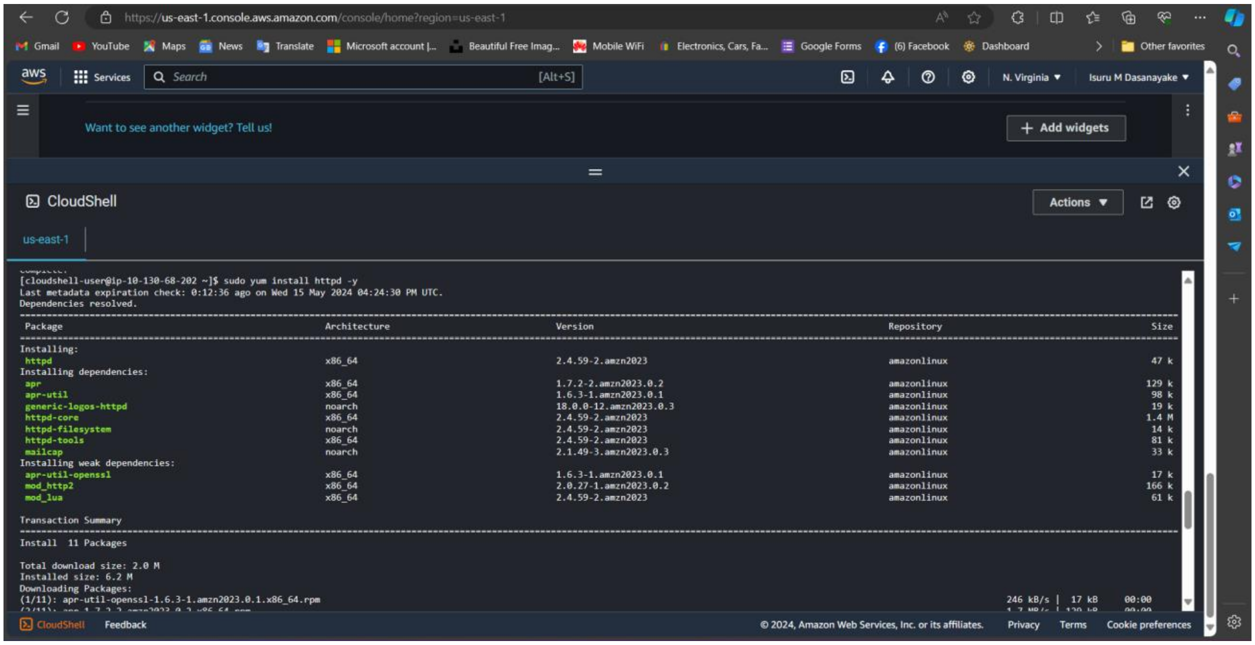

https://us-east-1.console.aws.amazon.com/console/home?region=us-east-1
Services
Search
[Alt+S]
N. Virginia
Isuru M Dasanayake
Want to see another widget? Tell us!
Add widgets
CloudShell
Actions
us-east-1
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

```
[cloudshell-user@ip-10-130-68-202 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:12:36 ago on Wed 15 May 2024 04:24:30 PM UTC.
Dependencies resolved.
================================================================================================================
 Package                    Architecture         Version                        Repository                 Size
================================================================================================================
Installing:
 httpd                      x86_64               2.4.59-2.amzn2023              amazonlinux                47 k
Installing dependencies:
 apr                        x86_64               1.7.2-2.amzn2023.0.2           amazonlinux               129 k
 apr-util                   x86_64               1.6.3-1.amzn2023.0.1           amazonlinux                98 k
 generic-logos-httpd        noarch               18.0.0-12.amzn2023.0.3         amazonlinux                19 k
 httpd-core                 x86_64               2.4.59-2.amzn2023              amazonlinux               1.4 M
 httpd-filesystem           noarch               2.4.59-2.amzn2023              amazonlinux                14 k
 httpd-tools                x86_64               2.4.59-2.amzn2023              amazonlinux                81 k
 mailcap                    noarch               2.1.49-3.amzn2023.0.3          amazonlinux                33 k
Installing weak dependencies:
 apr-util-openssl           x86_64               1.6.3-1.amzn2023.0.1           amazonlinux                17 k
 mod_http2                  x86_64               2.0.27-1.amzn2023.0.2          amazonlinux               166 k
 mod_lua                    x86_64               2.4.59-2.amzn2023              amazonlinux                61 k

Transaction Summary
================================================================================================================
Install  11 Packages

Total download size: 2.0 M
Installed size: 6.2 M
Downloading Packages:
(1/11): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm                               246 kB/s |  17 kB     00:00
```

1.07 Figures Transaction Summary



Want to see another widget? Tell us!

```
[cloudshell-user@ip-10-130-68-202 ~]$ sudo yum install httpd -y
Last metadata expiration check: 0:12:36 ago on Wed 15 May 2024 04:24:30 PM UTC.
Dependencies resolved.
==========================================================================================================================================
 Package                         Architecture          Version                           Repository                           Size
==========================================================================================================================================
Installing:
 httpd                           x86_64                2.4.59-2.amzn2023                 amazonlinux                          47 k
Installing dependencies:
 apr                             x86_64                1.7.2-2.amzn2023.0.2              amazonlinux                         129 k
 apr-util                        x86_64                1.6.3-1.amzn2023.0.1              amazonlinux                          98 k
 generic-logos-httpd             noarch                18.0.0-12.amzn2023.0.3            amazonlinux                          19 k
 httpd-core                      x86_64                2.4.59-2.amzn2023                 amazonlinux                         1.4 M
 httpd-filesystem                noarch                2.4.59-2.amzn2023                 amazonlinux                          14 k
 httpd-tools                     x86_64                2.4.59-2.amzn2023                 amazonlinux                          81 k
 mailcap                         noarch                2.1.49-3.amzn2023.0.3             amazonlinux                          33 k
Installing weak dependencies:
 apr-util-openssl                x86_64                1.6.3-1.amzn2023.0.1              amazonlinux                          17 k
 mod_http2                       x86_64                2.0.27-1.amzn2023.0.2             amazonlinux                         166 k
 mod_lua                         x86_64                2.4.59-2.amzn2023                 amazonlinux                          61 k

Transaction Summary
==========================================================================================================================================
Install  11 Packages

Total download size: 2.0 M
Installed size: 6.2 M
Downloading Packages:
(1/11): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm                                             246 kB/s |  17 kB     00:00
```

1.08 Figures Install httpd

CONFIGURE THE S3 BUCKET

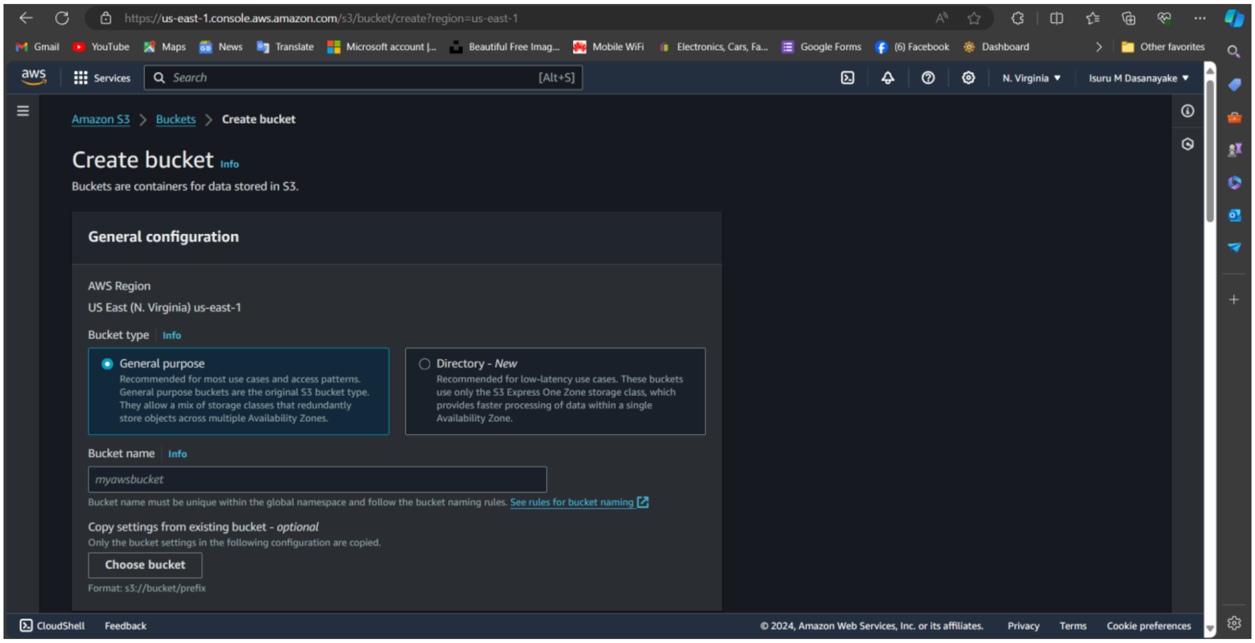
Setting up the S3 bucket is a crucial step in establishing your personal blog, facilitating efficient storage and retrieval of static files like images. Begin by logging into the AWS Management Console and accessing the S3 service. Initiate the creation of a new bucket by clicking the "Create bucket" button, where you'll specify a unique name and select the region that aligns with your requirements. Generally, sticking to default settings is advisable unless specific needs for versioning or logging arise. Upon bucket creation, you'll have a secure and scalable storage solution primed for integration with your EC2 instance.

Once the bucket is established, proceed to upload your static files. Navigate to the newly created bucket, then utilize the "Upload" button to add images and other static content. You can opt to drag and drop files or use the file picker to select them from your local system. Following the upload, each file will possess a unique URL, essential for embedding them in your HTML code. For instance, a file named `profile.jpg` uploaded to the bucket will sport a URL structured as `https://your-bucket-name.s3.amazonaws.com/profile.jpg`, enabling efficient content delivery.

To render your static files publicly accessible, adjust the bucket's permissions. Navigate to the "Permissions" tab of your S3 bucket and configure settings to allow public read access to the files. This can be executed by tweaking the bucket policy or utilizing the S3 console to set appropriate permissions. Exercise caution with these settings to expose only necessary components for your website. Once permissions are configured, your static files will be reachable via their URLs, seamlessly integrating into your HTML website hosted on the EC2 instance. This setup harnesses AWS S3's scalability and cost-effectiveness for static file delivery.

CONFIGURE THE S3 BUCKET

- **Create an S3 bucket**



Create bucket

Buckets are containers for data stored in S3.

General configuration

AWS Region

US East (N. Virginia) us-east-1

Bucket type

● General purpose
Recommended for most use cases and access patterns. General purpose buckets are the original S3 bucket type. They allow a mix of storage classes that redundantly store objects across multiple Availability Zones.

○ Directory - *New*
Recommended for low-latency use cases. These buckets use only the S3 Express One Zone storage class, which provides faster processing of data within a single Availability Zone.

Bucket name

myawsbucket

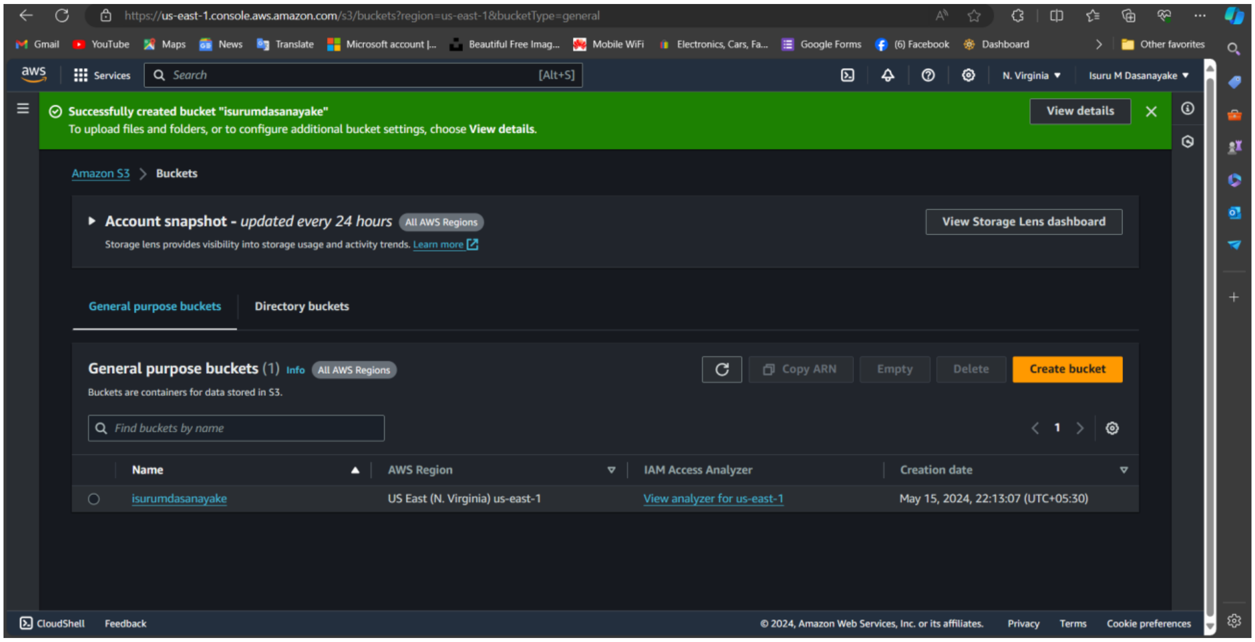
Bucket name must be unique within the global namespace and follow the bucket naming rules. See rules for bucket naming

Copy settings from existing bucket - *optional*

Only the bucket settings in the following configuration are copied.

Format: s3://bucket/prefix

1.09 Create S3 Bucket



Successfully created bucket "isurumdasanayake"
To upload files and folders, or to configure additional bucket settings, choose **View details**.

Account snapshot - *updated every 24 hours*

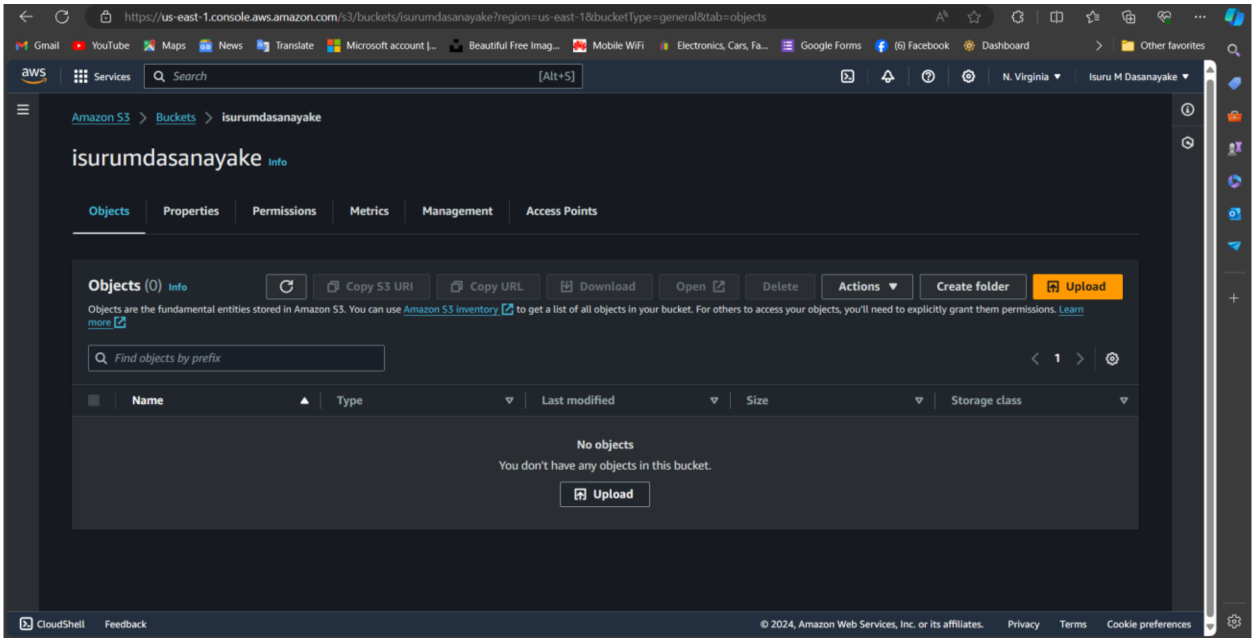
Storage lens provides visibility into storage usage and activity trends. Learn more

General purpose buckets (1)

Buckets are containers for data stored in S3.

Name	AWS Region	IAM Access Analyzer	Creation date
isurumdasanayake	US East (N. Virginia) us-east-1	View analyzer for us-east-1	May 15, 2024, 22:13:07 (UTC+05:30)

1.10 Figures General Purpose Bucket



isurumdasanayake Info

Objects (0) Info

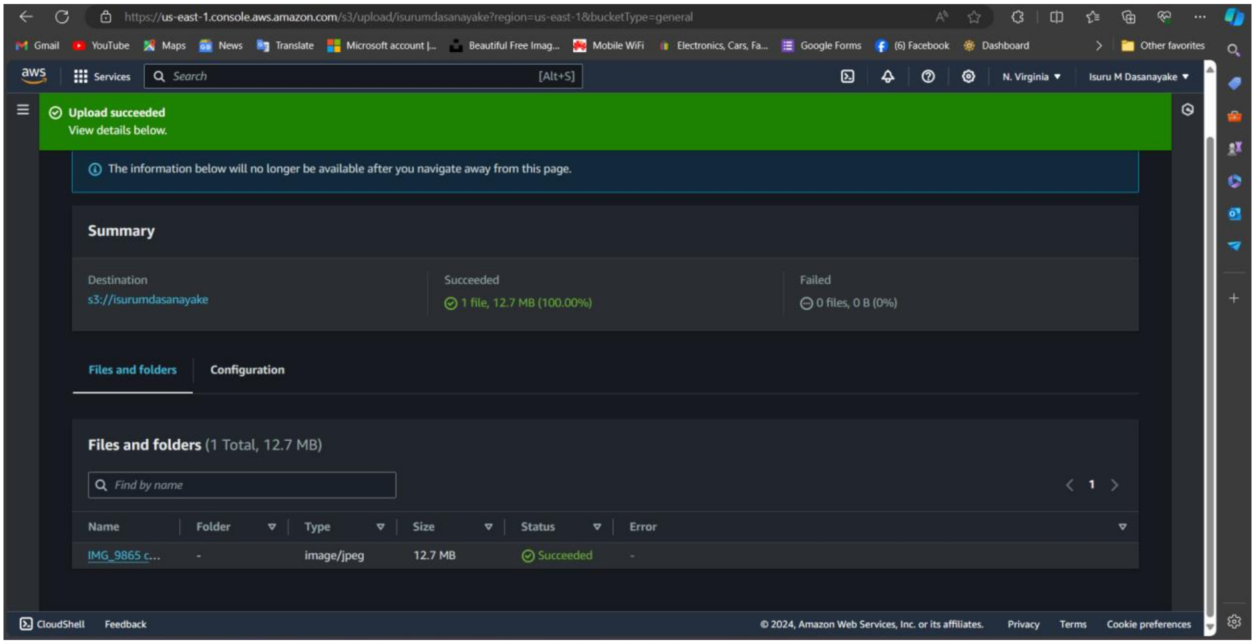
Objects are the fundamental entities stored in Amazon S3. You can use Amazon S3 inventory to get a list of all objects in your bucket. For others to access your objects, you'll need to explicitly grant them permissions. Learn more

Name	Type	Last modified	Size	Storage class

No objects
You don't have any objects in this bucket.

1.11 Figures Upload file in Bucket

- **Upload required image/s**


https://us-east-1.console.aws.amazon.com/s3/upload/isurumdasanayake?region=us-east-1&bucketType=general
Services
Search
[Alt+S]
N. Virginia
Isuru M Dasanayake
Files and folders
Configuration
Find by name
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

Upload succeeded
View details below.

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

Summary

Destination	Succeeded	Failed
s3://isurumdasanayake	1 file, 12.7 MB (100.00%)	0 files, 0 B (0%)

Files and folders (1 Total, 12.7 MB)

Name	Folder	Type	Size	Status	Error
IMG_9865 c...	-	image/jpeg	12.7 MB	Succeeded	-

1.12 Figures Upload succeeded in Bucket

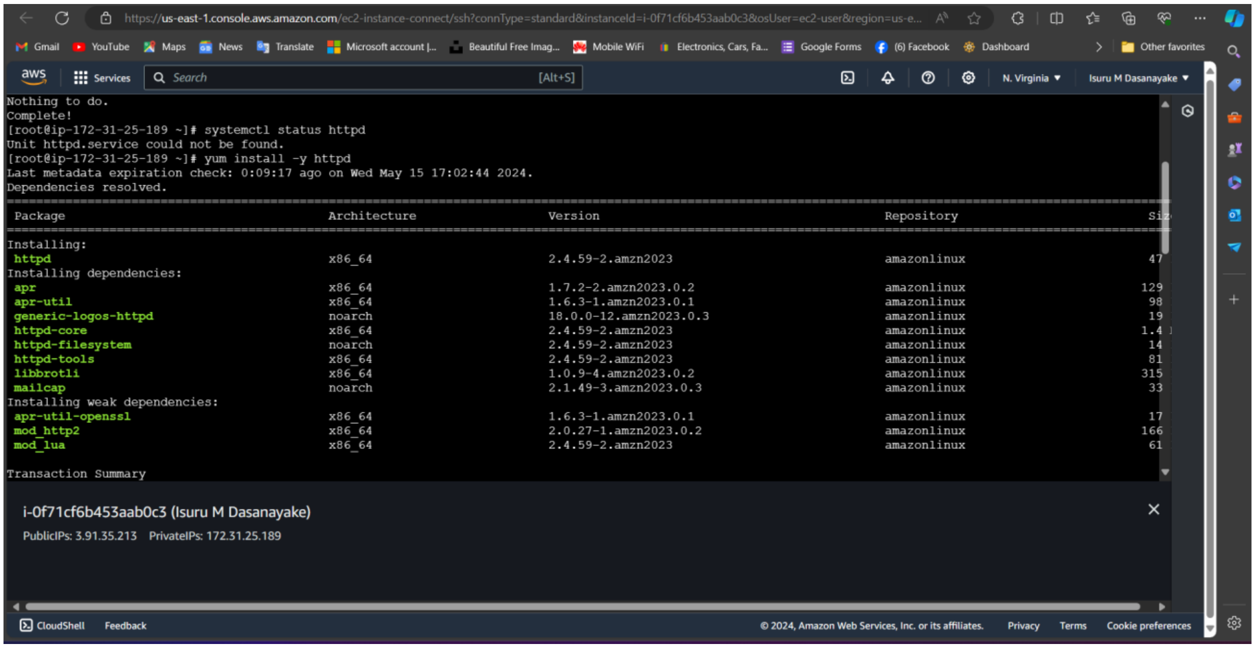
CREATE AND DEPLOY THE WEBSITE

Developing and deploying the website entails crafting HTML content and ensuring it's accurately hosted on the EC2 instance, with static assets linked from the S3 bucket. Begin by establishing an SSH connection to your EC2 instance and navigating to the Apache root directory, typically located at `/var/www/html`. Within this directory, create your primary HTML file, such as `index.html`, employing a text editor like `nano`. Within this HTML file, structure your webpage, incorporating images and other static content using the URLs provided by your S3 bucket. For instance, you may include an image using a tag like `<img src="https://my-personal-blog-bucket.s3.amazonaws.com/profile.jpg" alt="Profile Picture">`.

Once the HTML content is composed, save the file and ensure it's appropriately positioned in the Apache root directory. This placement guarantees that when users visit your EC2 instance's public IP address, the server serves the correct HTML content. To confirm the deployment, launch a web browser and enter the public IP address of your EC2 instance. If everything is configured accurately, your personal blog should load, showcasing the HTML content and images stored in the S3 bucket. This stage concludes the deployment process, rendering your blog publicly accessible and showcasing the seamless integration of AWS services for delivering a dynamic and scalable web application.

CREATE AND DEPLOY THE WEBSITE

- **Create the HTML file on the EC2 instance**



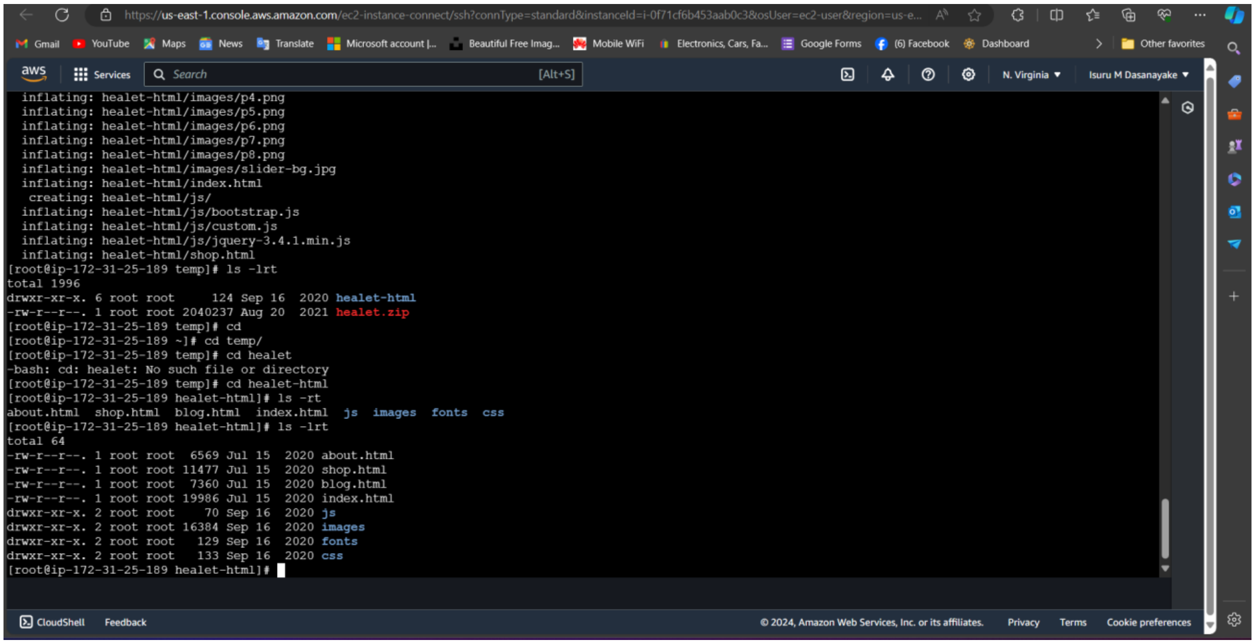
```
Nothing to do.
Complete!
[root@ip-172-31-25-189 ~]# systemctl status httpd
Unit httpd.service could not be found.
[root@ip-172-31-25-189 ~]# yum install -y httpd
Last metadata expiration check: 0:09:17 ago on Wed May 15 17:02:44 2024.
Dependencies resolved.
==============================================================================================================================
 Package                                    Architecture                 Version                                      Repository             Si
==============================================================================================================================
Installing:
 httpd                                      x86_64                       2.4.59-2.amzn2023                            amazonlinux            47
Installing dependencies:
 apr                                        x86_64                       1.7.2-2.amzn2023.0.2                         amazonlinux           129
 apr-util                                   x86_64                       1.6.3-1.amzn2023.0.1                         amazonlinux            98
 generic-logos-httpd                        noarch                       18.0.0-12.amzn2023.0.3                       amazonlinux            19
 httpd-core                                 x86_64                       2.4.59-2.amzn2023                            amazonlinux           1.4
 httpd-filesystem                           noarch                       2.4.59-2.amzn2023                            amazonlinux            14
 httpd-tools                                x86_64                       2.4.59-2.amzn2023                            amazonlinux            81
 libbrotli                                  x86_64                       1.0.9-4.amzn2023.0.2                         amazonlinux           315
 mailcap                                    noarch                       2.1.49-3.amzn2023.0.3                        amazonlinux            33
Installing weak dependencies:
 apr-util-openssl                           x86_64                       1.6.3-1.amzn2023.0.1                         amazonlinux            17
 mod_http2                                  x86_64                       2.0.27-1.amzn2023.0.2                        amazonlinux           166
 mod_lua                                    x86_64                       2.4.59-2.amzn2023                            amazonlinux            61

Transaction Summary
```

i-0f71cf6b453aab0c3 (Isuru M Dasanayake)

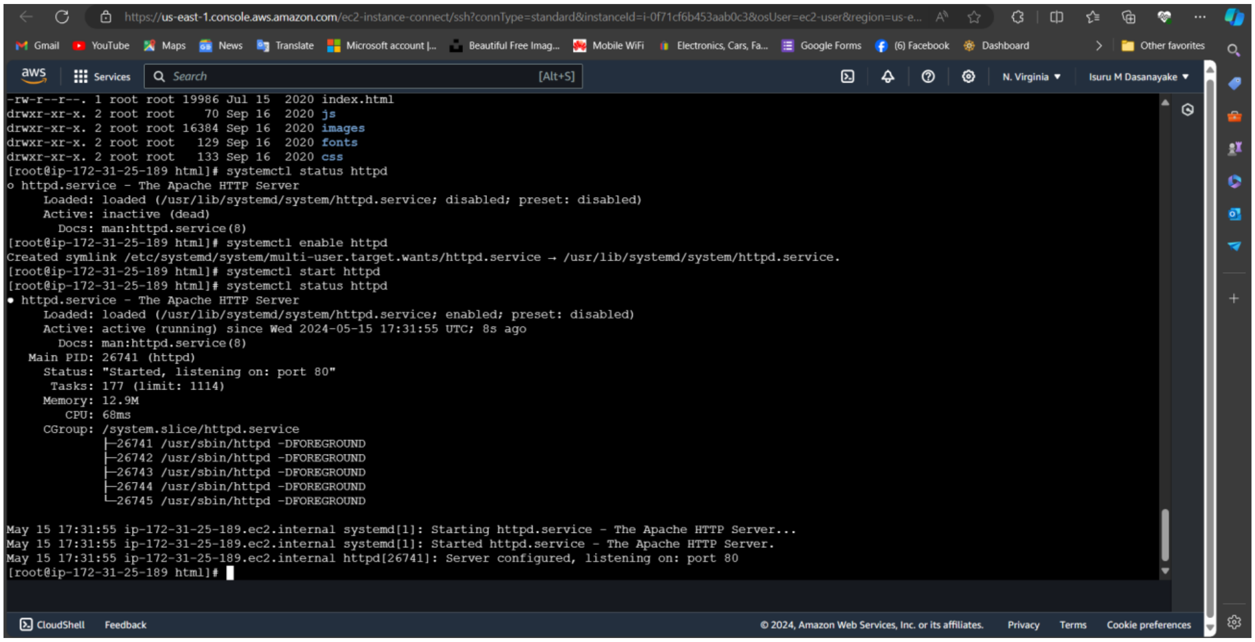
PublicIPs: 3.91.35.213 PrivateIPs: 172.31.25.189

1.13 Figures Yum Update -y



```
  inflating: healet-html/images/p4.png
  inflating: healet-html/images/p5.png
  inflating: healet-html/images/p6.png
  inflating: healet-html/images/p7.png
  inflating: healet-html/images/p8.png
  inflating: healet-html/images/slider-bg.jpg
  inflating: healet-html/index.html
   creating: healet-html/js/
  inflating: healet-html/js/bootstrap.js
  inflating: healet-html/js/custom.js
  inflating: healet-html/js/jquery-3.4.1.min.js
  inflating: healet-html/shop.html
[root@ip-172-31-25-189 temp]# ls -lrt
total 1996
drwxr-xr-x. 6 root root     124 Sep 16  2020 healet-html
-rw-r--r--. 1 root root 2040237 Aug 20  2021 healet.zip
[root@ip-172-31-25-189 temp]# cd
[root@ip-172-31-25-189 ~]# cd temp/
[root@ip-172-31-25-189 temp]# cd healet
-bash: cd: healet: No such file or directory
[root@ip-172-31-25-189 temp]# cd healet-html
[root@ip-172-31-25-189 healet-html]# ls -rt
about.html  shop.html  blog.html  index.html  js  images  fonts  css
[root@ip-172-31-25-189 healet-html]# ls -lrt
total 64
-rw-r--r--. 1 root root  6569 Jul 15  2020 about.html
-rw-r--r--. 1 root root 11477 Jul 15  2020 shop.html
-rw-r--r--. 1 root root  7360 Jul 15  2020 blog.html
-rw-r--r--. 1 root root 19986 Jul 15  2020 index.html
drwxr-xr-x. 2 root root    70 Sep 16  2020 js
drwxr-xr-x. 2 root root 16384 Sep 16  2020 images
drwxr-xr-x. 2 root root   129 Sep 16  2020 fonts
drwxr-xr-x. 2 root root   133 Sep 16  2020 css
[root@ip-172-31-25-189 healet-html]# 
```

1.14 Figures Unzip Zip folder

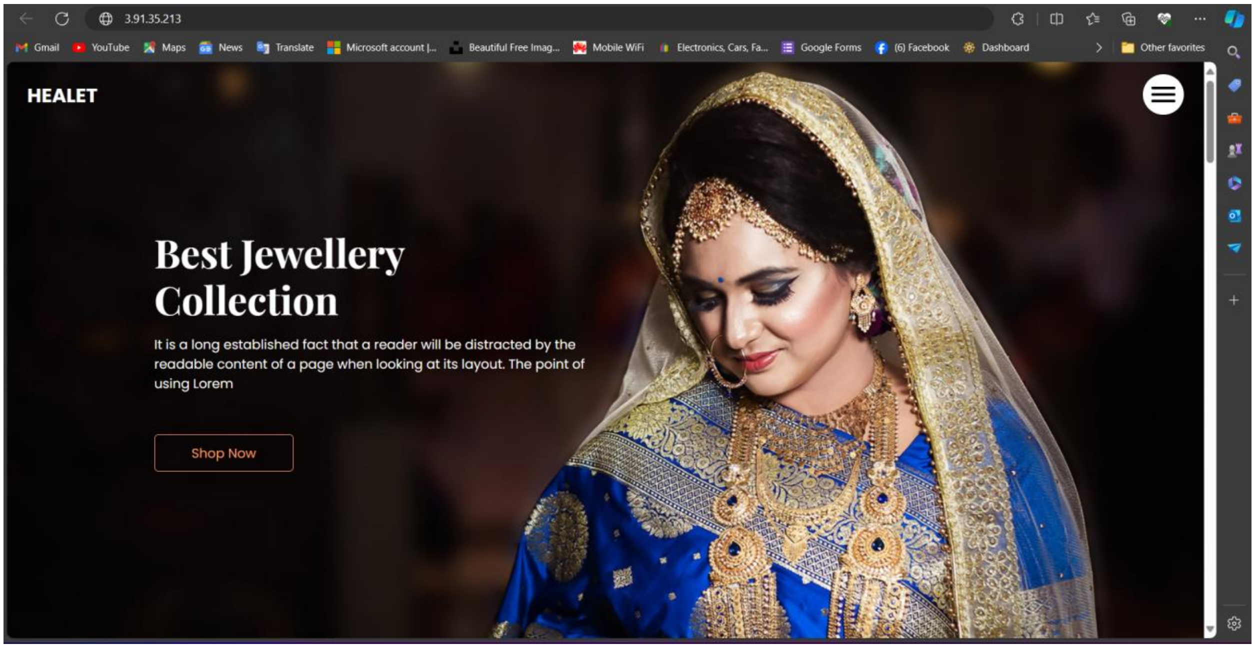


```
-rw-r--r--. 1 root root 19986 Jul 15  2020 index.html
drwxr-xr-x. 2 root root    70 Sep 16  2020 js
drwxr-xr-x. 2 root root 16384 Sep 16  2020 images
drwxr-xr-x. 2 root root   129 Sep 16  2020 fonts
drwxr-xr-x. 2 root root   133 Sep 16  2020 css
[root@ip-172-31-25-189 html]# systemctl status httpd
o httpd.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; preset: disabled)
     Active: inactive (dead)
       Docs: man:httpd.service(8)
[root@ip-172-31-25-189 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-25-189 html]# systemctl start httpd
[root@ip-172-31-25-189 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 Wed 2024-05-15 17:31:55 UTC; 8s ago
       Docs: man:httpd.service(8)
   Main PID: 26741 (httpd)
     Status: "Started, listening on: port 80"
      Tasks: 177 (limit: 1114)
     Memory: 12.9M
        CPU: 68ms
     CGroup: /system.slice/httpd.service
             ├─26741 /usr/sbin/httpd -DFOREGROUND
             ├─26742 /usr/sbin/httpd -DFOREGROUND
             ├─26743 /usr/sbin/httpd -DFOREGROUND
             ├─26744 /usr/sbin/httpd -DFOREGROUND
             └─26745 /usr/sbin/httpd -DFOREGROUND

May 15 17:31:55 ip-172-31-25-189.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
May 15 17:31:55 ip-172-31-25-189.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
May 15 17:31:55 ip-172-31-25-189.ec2.internal httpd[26741]: Server configured, listening on: port 80
[root@ip-172-31-25-189 html]# █
```

https://us-east-1.console.aws.amazon.com/ec2-instance-connect/ssh?connType=standard&instanceId=i-0f71cf6b453aab0c3&osUser=ec2-user&region=us-e...
aws
Services
Search
[Alt+S]
N. Virginia
Isuru M Dasanayake
CloudShell
Feedback
© 2024, Amazon Web Services, Inc. or its affiliates.
Privacy
Terms
Cookie preferences

1.15 Figures Unzipping Zip folder



HEALET

Best Jewellery Collection

It is a long established fact that a reader will be distracted by the readable content of a page when looking at its layout. The point of using Lorem

Shop Now

1.16 Figures Created Web Site