Adv. Devops Experiment no. 1

Name: Rohan Lalchandani
Class: D15A Roll no.: 25

Aim:
Part A) To host static and dynamic website on AWS and EC2.
Part B) To understand the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.

Theory:

AWS EC2 (Elastic Compute Cloud)

1. What is it? AWS EC2 provides scalable virtual servers (instances) in the cloud. You can use EC2 to run applications and services without managing physical hardware.

2. Key Features:

- **Instance Types:** Various types optimized for compute, memory, storage, or GPU capabilities.
- **Auto Scaling:** Automatically adjusts the number of instances based on traffic.
- **Elastic Load Balancing (ELB):** Distributes incoming traffic across multiple instances.
- **Security:** Integrates with AWS Identity and Access Management (IAM) and provides security groups and network ACLs.
- **Pricing:** Pay-as-you-go or reserved instances for long-term commitments.

3. Common Use Cases:

- Hosting websites and web applications.
- Running big data analytics.
- Building development and test environments.
- Running high-performance computing (HPC) applications.

AWS S3 (Simple Storage Service)

1. What is it? AWS S3 is an object storage service that provides highly scalable and durable storage for a wide variety of data types.

2. Key Features:

- **Storage Classes:** Multiple classes like Standard, Intelligent-Tiering, One Zone-IA, Glacier for different use cases and cost savings.
- **Data Durability:** Designed for 99.999999999% durability over a given year.
- **Scalability:** Automatically scales to accommodate data growth.
- **Security:** Data encryption (at rest and in transit), and access control policies.
- **Versioning:** Keeps multiple versions of an object to recover from accidental deletions.

3. Common Use Cases:

- Backup and restore.
- Data archiving.
- Application data storage.
- Content distribution and hosting.

AWS Cloud9

1. What is it? AWS Cloud9 is a cloud-based Integrated Development Environment (IDE) that allows you to write, run, and debug code from a web browser.

2. Key Features:

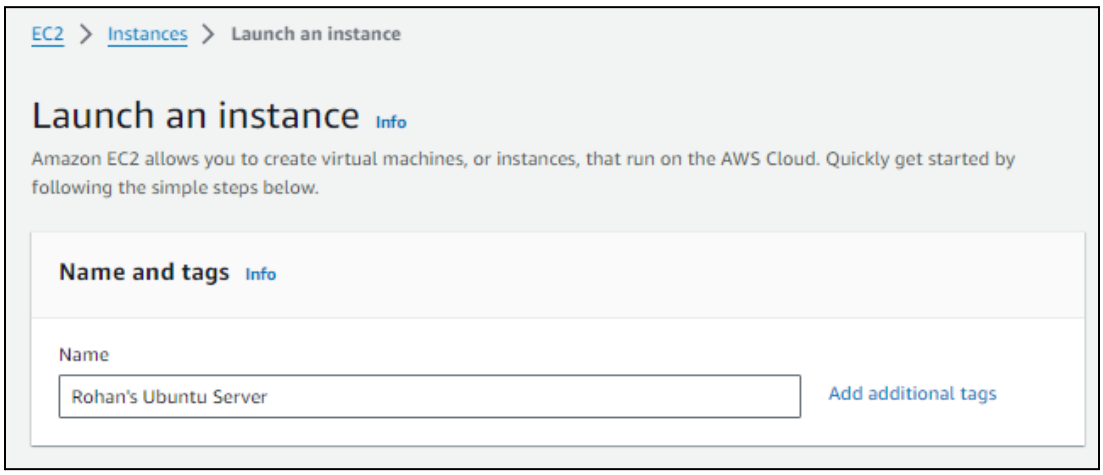
- **Environment:** Comes with pre-configured environments (Ubuntu-based) that include necessary tools and libraries.
- **Collaboration:** Multiple users can collaborate in real-time within the same IDE environment.
- **Integrated Tools:** Built-in terminal, debugger, and support for various programming languages.
- **AWS Integration:** Seamless integration with AWS services for deploying applications directly from the IDE.
- **Customizable:** Allows custom configurations and preferences for your development environment.

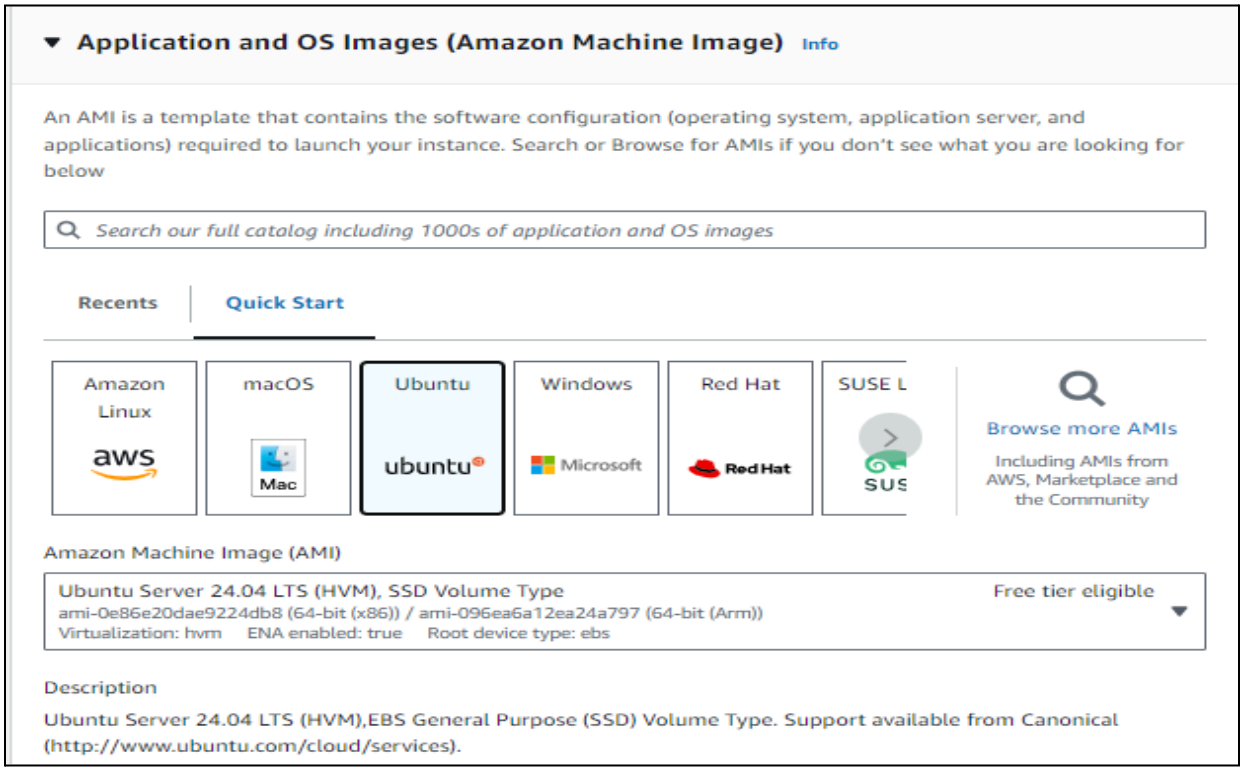
3. Common Use Cases:

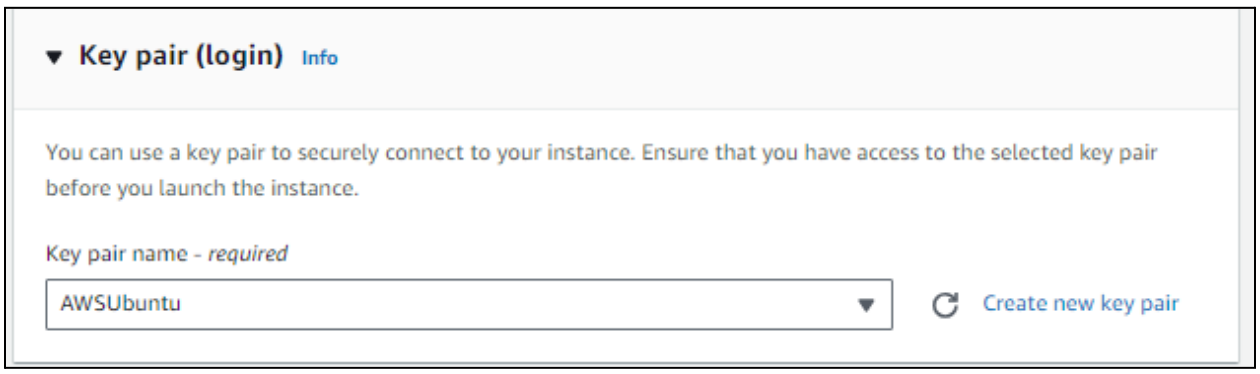
- Developing applications in a managed environment.
- Collaborative coding and debugging sessions.
- Learning and experimenting with code in a cloud-based IDE.

1. Static Website hosting using EC2(Ubuntu):

1) Instance creation and configuration:







Network settings Info

Edit

Network | Info

vpc-0f5e8abf0225b8a45

Subnet | Info

No preference (Default subnet in any availability zone)

Auto-assign public IP | Info

Enable

Additional charges apply when outside of free tier allowance

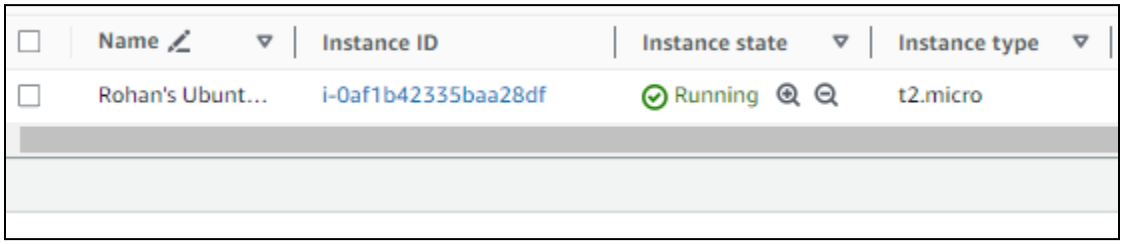
Firewall (security groups) | Info

A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.

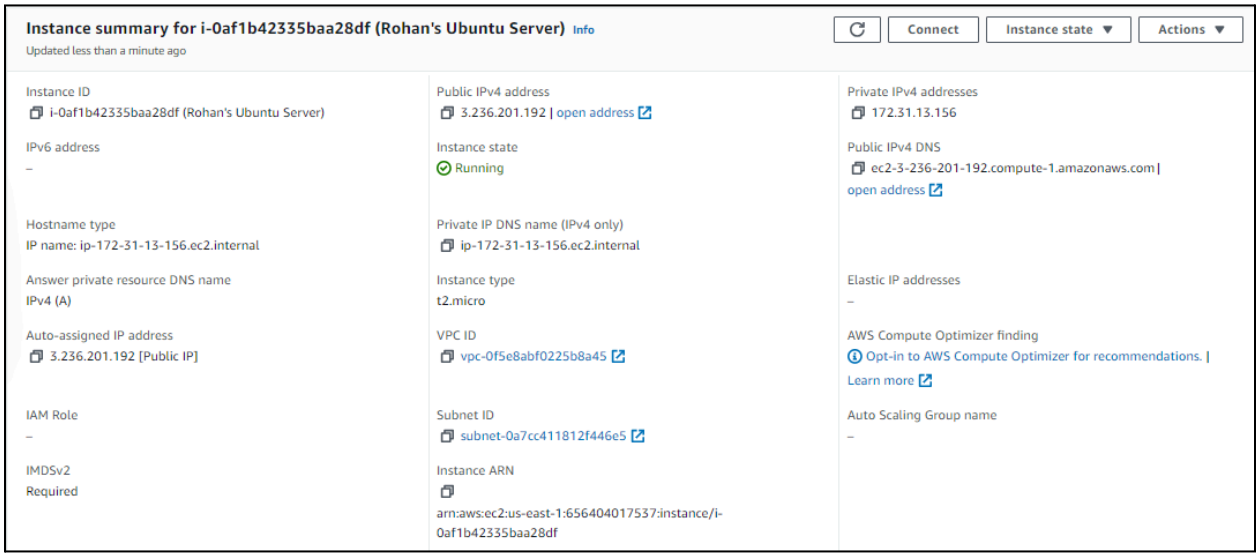
- (•) Create security group
- ( ) Select existing security group

We'll create a new security group called '**launch-wizard-3**' with the following rules:

- [x] Allow SSH traffic from
  Helps you connect to your instance
  Anywhere 0.0.0.0/0
- [ ] Allow HTTPS traffic from the internet
  To set up an endpoint, for example when creating a web server
- [ ] Allow HTTP traffic from the internet
  To set up an endpoint, for example when creating a web server



Name	Instance ID	Instance state	Instance type
Rohan's Ubunt...	i-0af1b42335baa28df	Running	t2.micro



Instance summary for i-0af1b42335baa28df (Rohan's Ubuntu Server) Info

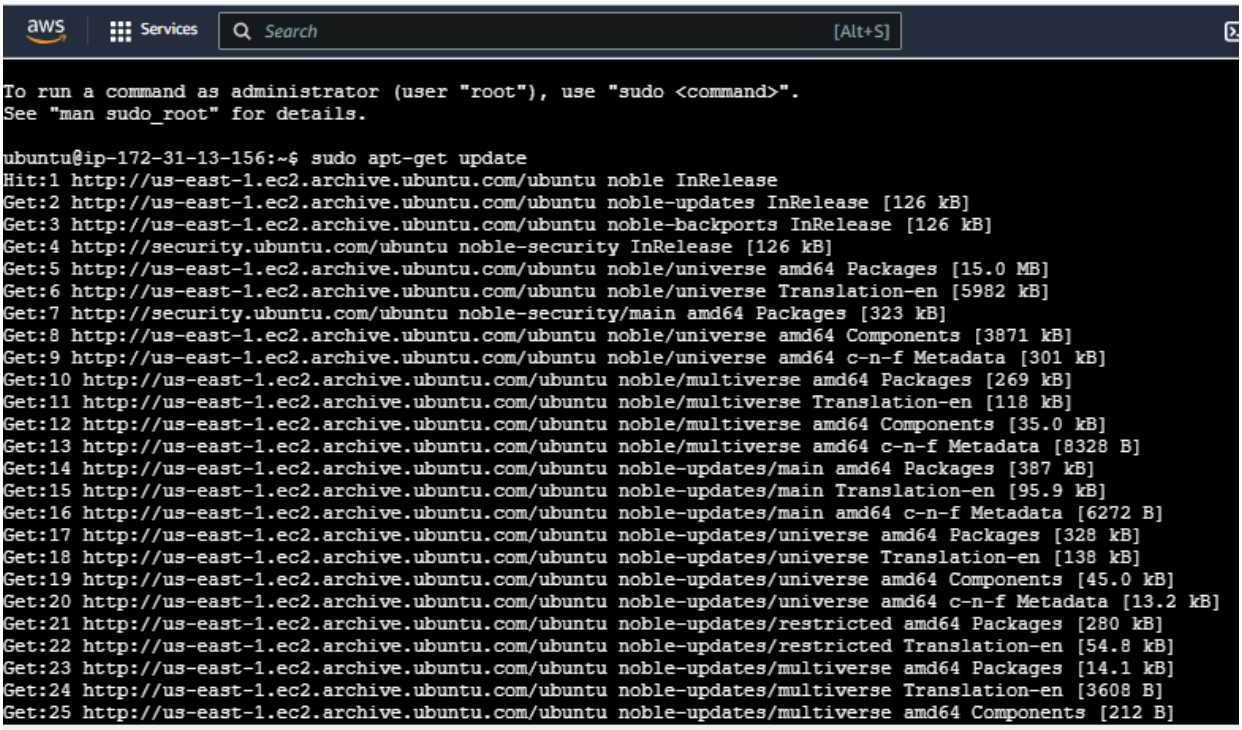
Updated less than a minute ago

Connect | Instance state | Actions

Instance ID i-0af1b42335baa28df (Rohan's Ubuntu Server)	Public IPv4 address 3.236.201.192 | open address	Private IPv4 addresses 172.31.13.156
IPv6 address –	Instance state Running	Public IPv4 DNS ec2-3-236-201-192.compute-1.amazonaws.com | open address
Hostname type IP name: ip-172-31-13-156.ec2.internal	Private IP DNS name (IPv4 only) ip-172-31-13-156.ec2.internal	
Answer private resource DNS name IPv4 (A)	Instance type t2.micro	Elastic IP addresses –
Auto-assigned IP address 3.236.201.192 [Public IP]	VPC ID vpc-0f5e8abf0225b8a45	AWS Compute Optimizer finding Opt-in to AWS Compute Optimizer for recommendations. | Learn more
IAM Role –	Subnet ID subnet-0a7cc411812f446e5	Auto Scaling Group name –
IMDSv2 Required	Instance ARN arn:aws:ec2:us-east-1:656404017537:instance/i-0af1b42335baa28df	

2) Connect to the instance and execute commands:

1. sudo apt-get update
2. sudo apt-get install apache2
3. systemctl status apache2

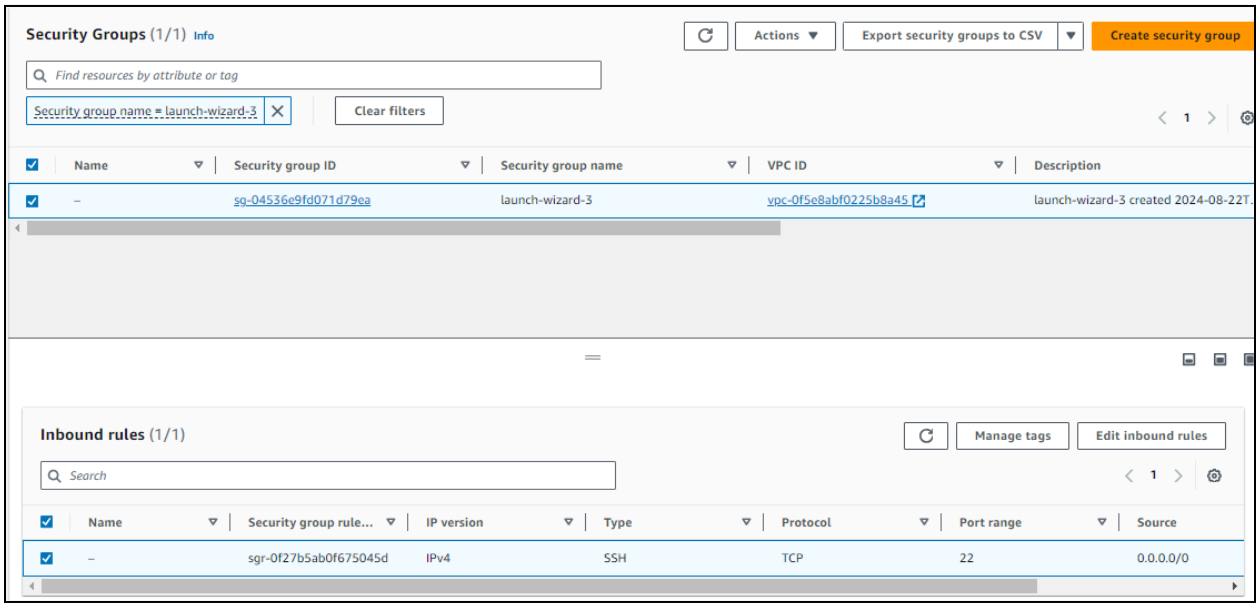


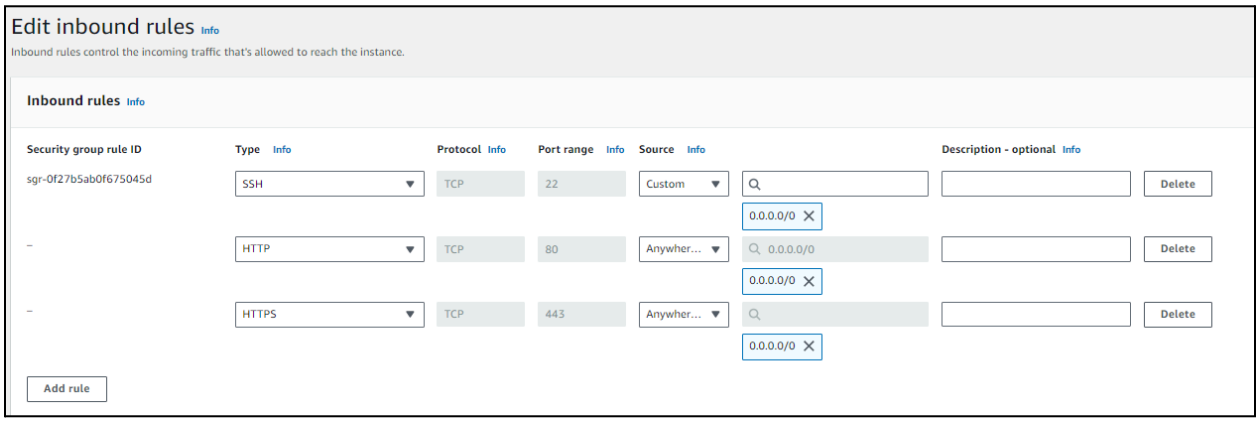

```
ubuntu@ip-172-31-13-156:~$ sudo apt-get install apache2
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1t64 libaprutil1-dbd-sqlite3 libaprutil1-ldap libaprutil1t64 liblua5.4-0 ssl-cert
0 upgraded, 10 newly installed, 0 to remove and 55 not upgraded.
Need to get 2083 kB of archives.
After this operation, 8094 kB of additional disk space will be used.
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libapr1t64 amd64 1.7.2-3.1build2 [107 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.6.3-1.1ubuntu7 [91.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [11.2 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap amd64 1.6.3-1.1ubuntu7 [9116 B]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 amd64 5.4.6-3build2 [166 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin amd64 2.4.58-1ubuntu8.4 [1329 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-1ubuntu8.4 [163 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu8.4 [97.1 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 amd64 2.4.58-1ubuntu8.4 [90.2 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 ssl-cert all 1.1.2ubuntu1 [17.8 kB]
Fetched 2083 kB in 0s (25.3 MB/s)
```

```
Do you want to continue? [Y/n] y
Get:1 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libapr1t64 amd64 1.7.2-3.1build2 [107 kB]
Get:2 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1t64 amd64 1.6.3-1.1ubuntu7 [91.9 kB]
Get:3 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-dbd-sqlite3 amd64 1.6.3-1.1ubuntu7 [11.2 kB]
Get:4 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 libaprutil1-ldap amd64 1.6.3-1.1ubuntu7 [9116 B]
Get:5 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 liblua5.4-0 amd64 5.4.6-3build2 [166 kB]
Get:6 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-bin amd64 2.4.58-1ubuntu8.4 [1329 kB]
Get:7 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-data all 2.4.58-1ubuntu8.4 [163 kB]
Get:8 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2-utils amd64 2.4.58-1ubuntu8.4 [97.1 kB]
Get:9 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 apache2 amd64 2.4.58-1ubuntu8.4 [90.2 kB]
Get:10 http://us-east-1.ec2.archive.ubuntu.com/ubuntu noble/main amd64 ssl-cert all 1.1.2ubuntu1 [17.8 kB]
Fetched 2083 kB in 0s (25.3 MB/s)
Preconfiguring packages ...
Selecting previously unselected package libapr1t64:amd64.
(Reading database ... 67741 files and directories currently installed.)
Preparing to unpack .../0-libapr1t64_1.7.2-3.1build2_amd64.deb ...
Unpacking libapr1t64:amd64 (1.7.2-3.1build2) ...
Selecting previously unselected package libaprutil1t64:amd64.
Preparing to unpack .../1-libaprutil1t64_1.6.3-1.1ubuntu7_amd64.deb ...
Unpacking libaprutil1t64:amd64 (1.6.3-1.1ubuntu7) ...
Selecting previously unselected package libaprutil1-dbd-sqlite3:amd64.
Preparing to unpack .../2-libaprutil1-dbd-sqlite3_1.6.3-1.1ubuntu7_amd64.deb ...
Unpacking libaprutil1-dbd-sqlite3:amd64 (1.6.3-1.1ubuntu7) ...
Selecting previously unselected package libaprutil1-ldap:amd64.
Preparing to unpack .../3-libaprutil1-ldap_1.6.3-1.1ubuntu7_amd64.deb ...
```

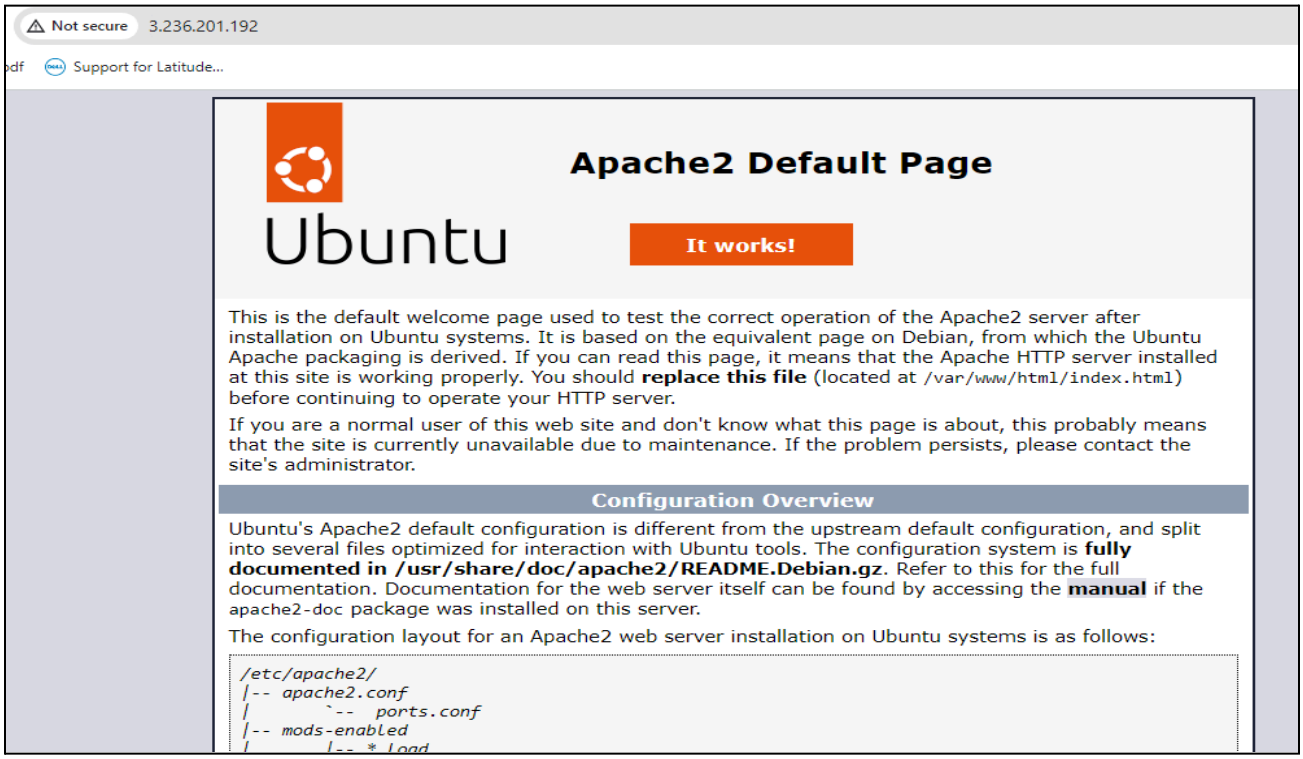
```
ubuntu@ip-172-31-13-156:~$ systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Thu 2024-08-22 17:23:12 UTC; 11min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 2399 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.4M (peak: 5.6M)
        CPU: 70ms
     CGroup: /system.slice/apache2.service
             ├─2399 /usr/sbin/apache2 -k start
             ├─2402 /usr/sbin/apache2 -k start
             └─2403 /usr/sbin/apache2 -k start

Aug 22 17:23:12 ip-172-31-13-156 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Aug 22 17:23:12 ip-172-31-13-156 systemd[1]: Started apache2.service - The Apache HTTP Server.
ubuntu@ip-172-31-13-156:~$ sudo su
root@ip-172-31-13-156:/home/ubuntu# cd /var/ww/html/
bash: cd: /var/ww/html/: No such file or directory
root@ip-172-31-13-156:/home/ubuntu# cd /var/www/html/
root@ip-172-31-13-156:/var/www/html#
```

3) Editing the inbound rules:



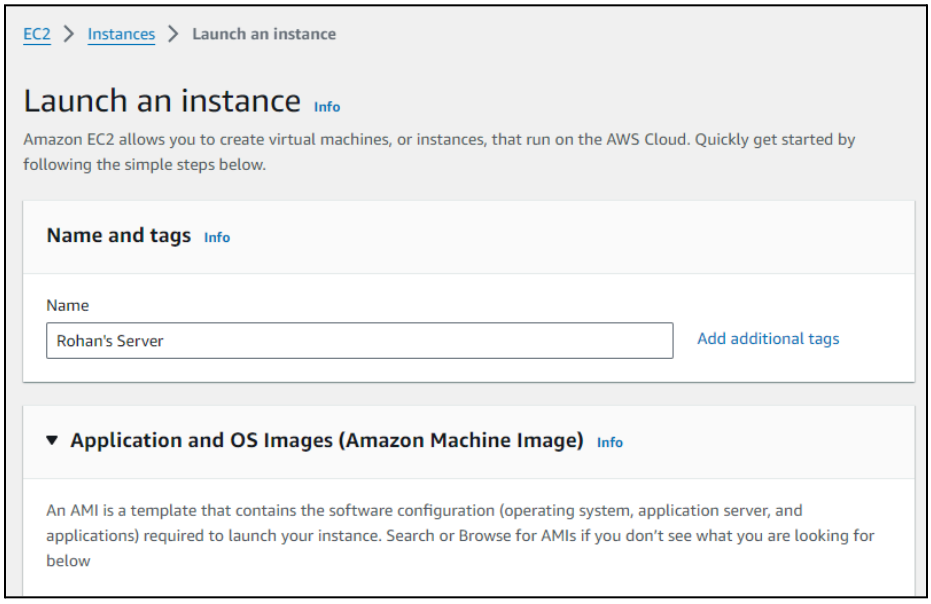


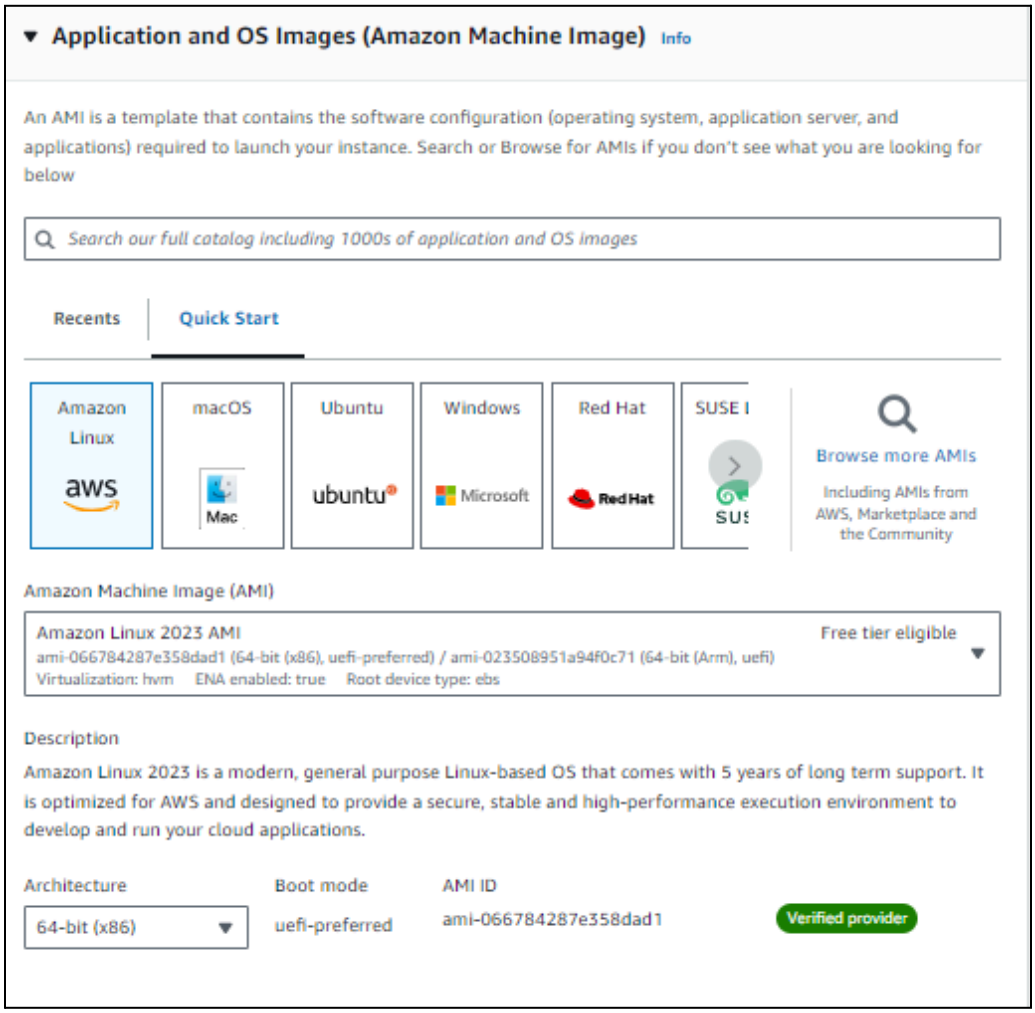
4) Go to the public IP link in new tab.

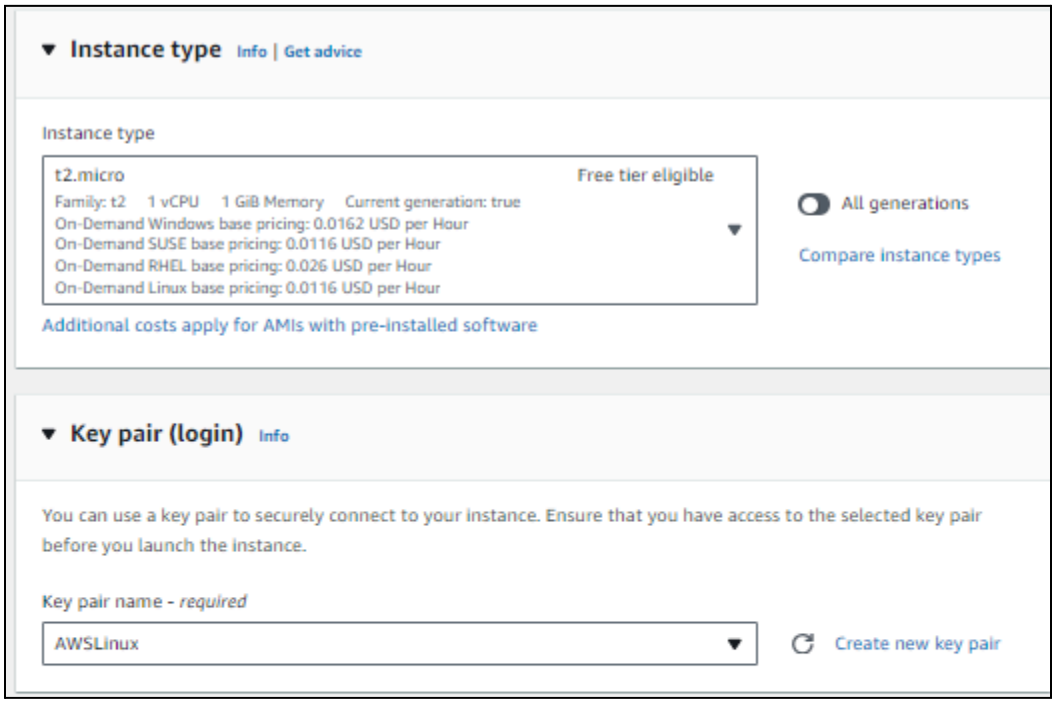


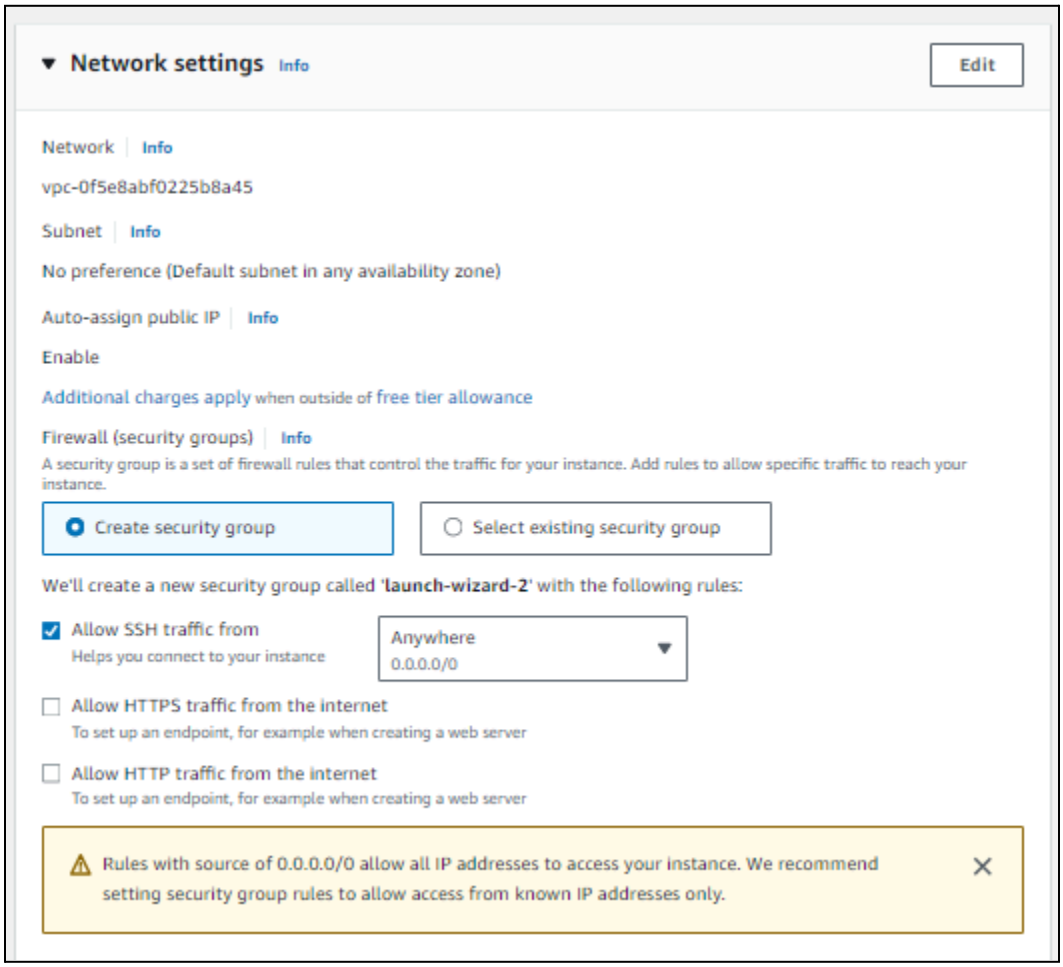
1. Dynamic hosting using EC2(Amazon Linux):

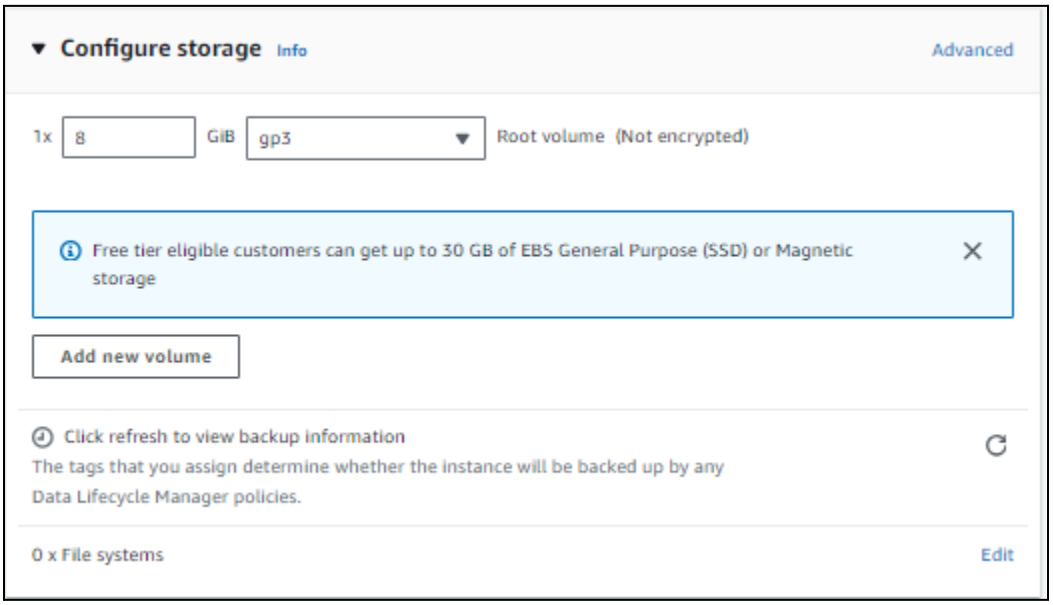
1) Instance creation:




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
Recents
Quick Start
Amazon Linux
macOS
Ubuntu
Windows
Red Hat
SUSE I
Browse more AMIs
Including AMIs from AWS, Marketplace and the Community
Amazon Machine Image (AMI)
Amazon Linux 2023 AMI
ami-066784287e358dad1 (64-bit (x86), uefi-preferred) / ami-023508951a94f0c71 (64-bit (Arm), uefi)
Virtualization: hvm ENA enabled: true Root device type: ebs
Free tier eligible
Description
Amazon Linux 2023 is a modern, general purpose Linux-based OS that comes with 5 years of long term support. It is optimized for AWS and designed to provide a secure, stable and high-performance execution environment to develop and run your cloud applications.
Architecture
64-bit (x86)
Boot mode
uefi-preferred
AMI ID
ami-066784287e358dad1
Verified provider


Instance type Info | Get advice
Instance type
t2.micro
Free tier eligible
Family: t2 1 vCPU 1 GiB Memory Current generation: true
On-Demand Windows base pricing: 0.0162 USD per Hour
On-Demand SUSE base pricing: 0.0116 USD per Hour
On-Demand RHEL base pricing: 0.026 USD per Hour
On-Demand Linux base pricing: 0.0116 USD per Hour
All generations
Compare instance types
Additional costs apply for AMIs with pre-installed software
Key pair (login) Info
You can use a key pair to securely connect to your instance. Ensure that you have access to the selected key pair before you launch the instance.
Key pair name - required
AWSLinux
Create new key pair


Network settings Info
Edit
Network Info
vpc-0f5e8abf0225b8a45
Subnet Info
No preference (Default subnet in any availability zone)
Auto-assign public IP Info
Enable
Additional charges apply when outside of free tier allowance
Firewall (security groups) Info
A security group is a set of firewall rules that control the traffic for your instance. Add rules to allow specific traffic to reach your instance.
Create security group
Select existing security group
We'll create a new security group called 'launch-wizard-2' with the following rules:
Allow SSH traffic from
Helps you connect to your instance
Anywhere
0.0.0.0/0
Allow HTTPS traffic from the internet
To set up an endpoint, for example when creating a web server
Allow HTTP traffic from the internet
To set up an endpoint, for example when creating a web server
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.


Configure storage Info
Advanced
1x
8
GiB
gp3
Root volume (Not encrypted)
Free tier eligible customers can get up to 30 GB of EBS General Purpose (SSD) or Magnetic storage
Add new volume
Click refresh to view backup information
The tags that you assign determine whether the instance will be backed up by any Data Lifecycle Manager policies.
0 x File systems
Edit

2) Connect to the instance:

Execute commands:

1. sudo su -
2. yum update -y
3. yum install -y httpd
4. systemctl status httpd
5. mkdir aws_assg3
6. cd aws_assg3
7. For this experiment we have created a website which we have uploaded on Github.com.
8. Copy the Download Link for the .zip file of the website.
9. using the wget command, download the zip file to the folder.
10. unzip the main.zip file and navigate into the "OnlinePrintingServices-main" folder using the cd command.
11. move all the contents from the folder to "/var/www/html/"

```
   ,     #_
   ~\_  ####_        Amazon Linux 2023
  ~~  \_#####\
  ~~     \###|
  ~~       \#/ ___   https://aws.amazon.com/linux/amazon-linux-2023
   ~~       V~' '->
    ~~~         /
      ~~._.   _/
         _/ _/
       _/m/'
Last login: Wed Aug 21 20:17:27 2024 from 18.206.107.28
[ec2-user@ip-172-31-51-115 ~]$ sudo su -
[root@ip-172-31-51-115 ~]# yum update -y
Last metadata expiration check: 0:10:51 ago on Wed Aug 21 20:12:36 2024.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-51-115 ~]# yum install -y httpd
Last metadata expiration check: 0:11:57 ago on Wed Aug 21 20:12:36 2024.
Dependencies resolved.
================================================================================
 Package                 Architecture   Version                  Repository    Size
================================================================================
Installing:
 httpd                   x86_64         2.4.62-1.amzn2023         amazonlinux   48 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         2.4.62-1.amzn2023         amazonlinux  1.4 M
```

```
 httpd-core              x86_64         2.4.62-1.amzn2023         amazonlinux  1.4 M
 httpd-filesystem        noarch         2.4.62-1.amzn2023         amazonlinux   14 k
 httpd-tools             x86_64         2.4.62-1.amzn2023         amazonlinux   81 k
 libbrotli               x86_64         1.0.9-4.amzn2023.0.2      amazonlinux  315 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.amzn2023.0.3     amazonlinux  166 k
 mod_lua                 x86_64         2.4.62-1.amzn2023         amazonlinux   61 k

Transaction Summary
================================================================================
Install  12 Packages

Total download size: 2.3 M
Installed size: 6.9 M
Downloading Packages:
(1/12): apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64.rpm      256 kB/s |  17 kB   00:00
(2/12): apr-1.7.2-2.amzn2023.0.2.x86_64.rpm                   1.8 MB/s | 129 kB   00:00
(3/12): apr-util-1.6.3-1.amzn2023.0.1.x86_64.rpm              1.2 MB/s |  98 kB   00:00
(4/12): generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch.rpm 1.0 MB/s |  19 kB   00:00
(5/12): httpd-2.4.62-1.amzn2023.x86_64.rpm                    2.3 MB/s |  48 kB   00:00
(6/12): httpd-filesystem-2.4.62-1.amzn2023.noarch.rpm         657 kB/s |  14 kB   00:00
(7/12): httpd-core-2.4.62-1.amzn2023.x86_64.rpm                34 MB/s | 1.4 MB   00:00
(8/12): libbrotli-1.0.9-4.amzn2023.0.2.x86_64.rpm              14 MB/s | 315 kB   00:00
(9/12): mailcap-2.1.49-3.amzn2023.0.3.noarch.rpm              1.4 MB/s |  33 kB   00:00
(10/12): mod_http2-2.0.27-1.amzn2023.0.3.x86_64.rpm           5.9 MB/s | 166 kB   00:00
(11/12): mod_lua-2.4.62-1.amzn2023.x86_64.rpm                 3.5 MB/s |  61 kB   00:00
(12/12): httpd-tools-2.4.62-1.amzn2023.x86_64.rpm             708 kB/s |  81 kB   00:00
--------------------------------------------------------------------------------
```

```
--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------
Total                                                                                                                                    8.7 MB/s | 2.3 MB     00:00
Running transaction check
Transaction check succeeded.
Running transaction test
Transaction test succeeded.
Running transaction
  Preparing        :                                                                                                                                                  1/1
  Installing       : apr-1.7.2-2.amzn2023.0.2.x86_64                                                                                                                 1/12
  Installing       : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64                                                                                                    2/12
  Installing       : apr-util-1.6.3-1.amzn2023.0.1.x86_64                                                                                                            3/12
  Installing       : mailcap-2.1.49-3.amzn2023.0.3.noarch                                                                                                            4/12
  Installing       : httpd-tools-2.4.62-1.amzn2023.x86_64                                                                                                            5/12
  Installing       : libbrotli-1.0.9-4.amzn2023.0.2.x86_64                                                                                                           6/12
  Running scriptlet: httpd-filesystem-2.4.62-1.amzn2023.noarch                                                                                                       7/12
  Installing       : httpd-filesystem-2.4.62-1.amzn2023.noarch                                                                                                       7/12
  Installing       : httpd-core-2.4.62-1.amzn2023.x86_64                                                                                                             8/12
  Installing       : mod_http2-2.0.27-1.amzn2023.0.3.x86_64                                                                                                          9/12
  Installing       : mod_lua-2.4.62-1.amzn2023.x86_64                                                                                                               10/12
  Installing       : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch                                                                                              11/12
  Installing       : httpd-2.4.62-1.amzn2023.x86_64                                                                                                                 12/12
  Running scriptlet: httpd-2.4.62-1.amzn2023.x86_64                                                                                                                 12/12
  Verifying        : apr-1.7.2-2.amzn2023.0.2.x86_64                                                                                                                 1/12
  Verifying        : apr-util-1.6.3-1.amzn2023.0.1.x86_64                                                                                                            2/12
  Verifying        : apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64                                                                                                    3/12
  Verifying        : generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch                                                                                               4/12
  Verifying        : httpd-2.4.62-1.amzn2023.x86_64                                                                                                                  5/12
  Verifying        : httpd-core-2.4.62-1.amzn2023.x86_64                                                                                                             6/12
  Verifying        : httpd-filesystem-2.4.62-1.amzn2023.noarch                                                                                                       7/12
  Verifying        : httpd-tools-2.4.62-1.amzn2023.x86_64                                                                                                            8/12
```

```
  Verifying        : httpd-tools-2.4.62-1.amzn2023.x86_64                                                                                                            8/12
  Verifying        : libbrotli-1.0.9-4.amzn2023.0.2.x86_64                                                                                                           9/12
  Verifying        : mailcap-2.1.49-3.amzn2023.0.3.noarch                                                                                                           10/12
  Verifying        : mod_http2-2.0.27-1.amzn2023.0.3.x86_64                                                                                                         11/12
  Verifying        : mod_lua-2.4.62-1.amzn2023.x86_64                                                                                                               12/12

Installed:
  apr-1.7.2-2.amzn2023.0.2.x86_64          apr-util-1.6.3-1.amzn2023.0.1.x86_64    apr-util-openssl-1.6.3-1.amzn2023.0.1.x86_64    generic-logos-httpd-18.0.0-12.amzn2023.0.3.noarch
  httpd-2.4.62-1.amzn2023.x86_64           httpd-core-2.4.62-1.amzn2023.x86_64     httpd-filesystem-2.4.62-1.amzn2023.noarch       httpd-tools-2.4.62-1.amzn2023.x86_64
  libbrotli-1.0.9-4.amzn2023.0.2.x86_64    mailcap-2.1.49-3.amzn2023.0.3.noarch    mod_http2-2.0.27-1.amzn2023.0.3.x86_64          mod_lua-2.4.62-1.amzn2023.x86_64

Complete!
[root@ip-172-31-51-115 ~]# systemctl status httpd
○ 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-172-31-51-115 ~]# mkdir aws_assg3
[root@ip-172-31-51-115 ~]# cd aws_assg3
[root@ip-172-31-51-115 aws_assg3]# wget https://github.com/Rohan-Lalchandani08/OnlinePrintingServices.git
--2024-08-21 20:32:34--  https://github.com/Rohan-Lalchandani08/OnlinePrintingServices.git
Resolving github.com (github.com)... 140.82.113.4
Connecting to github.com (github.com)|140.82.113.4|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://github.com/Rohan-Lalchandani08/OnlinePrintingServices [following]
--2024-08-21 20:32:34--  https://github.com/Rohan-Lalchandani08/OnlinePrintingServices
Reusing existing connection to github.com:443.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'OnlinePrintingServices.git'
```

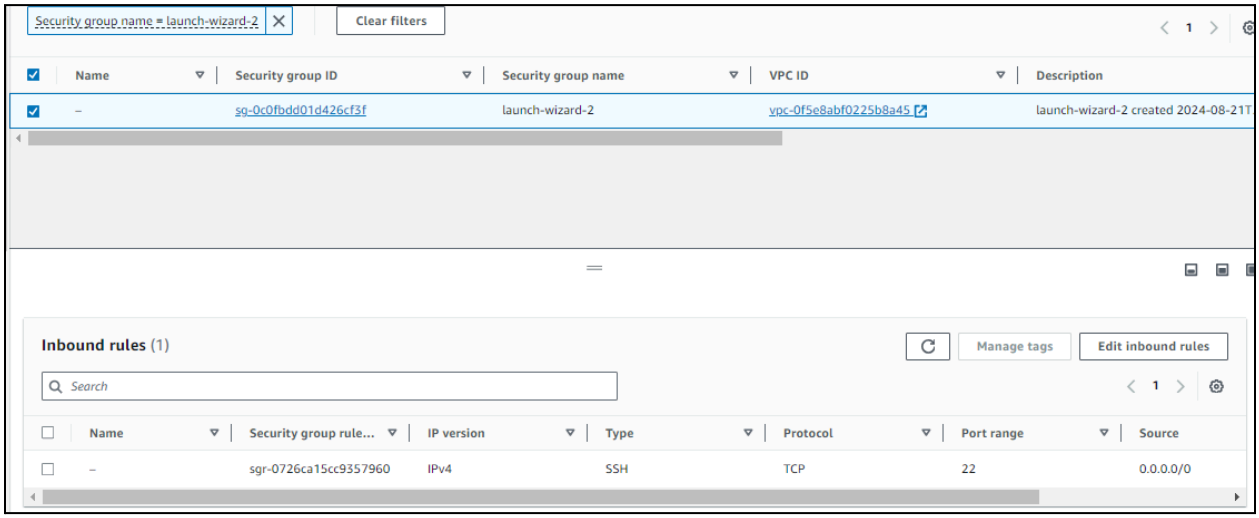
```
Saving to: 'OnlinePrintingServices.git'

OnlinePrintingServices.git                        [ <=>                                                                                   ] 277.94K  --.-KB/s    in 0.007s

2024-08-21 20:32:35 (39.4 MB/s) - 'OnlinePrintingServices.git' saved [284608]

[root@ip-172-31-51-115 aws_assg3]# ls -lrt
total 280
-rw-r--r--. 1 root root 284608 Aug 21 20:32 OnlinePrintingServices.git
[root@ip-172-31-51-115 aws_assg3]# wget https://github.com/Rohan-Lalchandani08/OnlinePrintingServices/archive/refs/heads/main.zip
--2024-08-21 20:35:21--  https://github.com/Rohan-Lalchandani08/OnlinePrintingServices/archive/refs/heads/main.zip
Resolving github.com (github.com)... 140.82.113.3
Connecting to github.com (github.com)|140.82.113.3|:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://codeload.github.com/Rohan-Lalchandani08/OnlinePrintingServices/zip/refs/heads/main [following]
--2024-08-21 20:35:21--  https://codeload.github.com/Rohan-Lalchandani08/OnlinePrintingServices/zip/refs/heads/main
Resolving codeload.github.com (codeload.github.com)... 140.82.113.9
Connecting to codeload.github.com (codeload.github.com)|140.82.113.9|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [application/zip]
Saving to: 'main.zip'

main.zip                                          [ <=>                                                                                   ] 450.42K  --.-KB/s    in 0.02s

2024-08-21 20:35:21 (28.4 MB/s) - 'main.zip' saved [461233]

[root@ip-172-31-51-115 aws_assg3]# ls -lrt
total 732
-rw-r--r--. 1 root root 284608 Aug 21 20:32 OnlinePrintingServices.git
-rw-r--r--. 1 root root 461233 Aug 21 20:35 main.zip
```

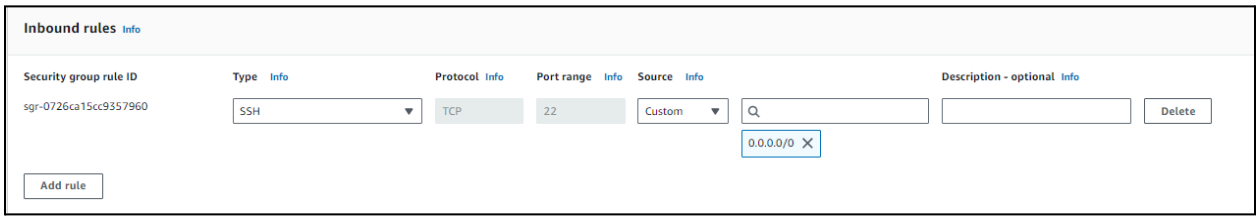
```
-rw-r--r--. 1 root root 284608 Aug 21 20:32 OnlinePrintingServices.git
-rw-r--r--. 1 root root 461233 Aug 21 20:35 main.zip
[root@ip-172-31-51-115 aws_assg3]# unzip main.zip
Archive:  main.zip
e89afae28997f8992f8d48da34c717b34da37c24
   creating: OnlinePrintingServices-main/
 extracting: OnlinePrintingServices-main/about.html
  inflating: OnlinePrintingServices-main/audio.mp3
 extracting: OnlinePrintingServices-main/banner.png
  inflating: OnlinePrintingServices-main/brochure.jpg
 extracting: OnlinePrintingServices-main/buisnesscard.png
  inflating: OnlinePrintingServices-main/contact.html
 extracting: OnlinePrintingServices-main/facebook.png
 extracting: OnlinePrintingServices-main/flyers.png
  inflating: OnlinePrintingServices-main/index.html
 extracting: OnlinePrintingServices-main/insta.png
 extracting: OnlinePrintingServices-main/poster.png
  inflating: OnlinePrintingServices-main/printer logo.png
  inflating: OnlinePrintingServices-main/x.png
[root@ip-172-31-51-115 aws_assg3]# ls -lrt
total 748
drwxr-xr-x. 2 root root  16384 Aug 20 04:47 OnlinePrintingServices-main
-rw-r--r--. 1 root root 284608 Aug 21 20:32 OnlinePrintingServices.git
-rw-r--r--. 1 root root 461233 Aug 21 20:35 main.zip
[root@ip-172-31-51-115 aws_assg3]# cd OnlinePrintingServices-main
[root@ip-172-31-51-115 OnlinePrintingServices-main]# ls -lrt
total 480
-rw-r--r--. 1 root root  14913 Aug 20 04:47  x.png
-rw-r--r--. 1 root root  17827 Aug 20 04:47 'printer logo.png'
-rw-r--r--. 1 root root  68827 Aug 20 04:47  poster.png
```

```
-rw-r--r--. 1 root root  17827 Aug 20 04:47 'printer logo.png'
-rw-r--r--. 1 root root  68827 Aug 20 04:47  poster.png
-rw-r--r--. 1 root root  28213 Aug 20 04:47  insta.png
-rw-r--r--. 1 root root   2821 Aug 20 04:47  index.html
-rw-r--r--. 1 root root 102141 Aug 20 04:47  flyers.png
-rw-r--r--. 1 root root  11110 Aug 20 04:47  facebook.png
-rw-r--r--. 1 root root    936 Aug 20 04:47  contact.html
-rw-r--r--. 1 root root  34395 Aug 20 04:47  buisnesscard.png
-rw-r--r--. 1 root root  86675 Aug 20 04:47  brochure.jpg
-rw-r--r--. 1 root root  69398 Aug 20 04:47  banner.png
-rw-r--r--. 1 root root  29997 Aug 20 04:47  audio.mp3
-rw-r--r--. 1 root root      2 Aug 20 04:47  about.html
[root@ip-172-31-51-115 OnlinePrintingServices-main]# mv * /var/www/html/
[root@ip-172-31-51-115 OnlinePrintingServices-main]# cd /var/www/html/
[root@ip-172-31-51-115 html]# ls -lrt
total 480
-rw-r--r--. 1 root root  14913 Aug 20 04:47  x.png
-rw-r--r--. 1 root root  17827 Aug 20 04:47 'printer logo.png'
-rw-r--r--. 1 root root  68827 Aug 20 04:47  poster.png
-rw-r--r--. 1 root root  28213 Aug 20 04:47  insta.png
-rw-r--r--. 1 root root   2821 Aug 20 04:47  index.html
-rw-r--r--. 1 root root 102141 Aug 20 04:47  flyers.png
-rw-r--r--. 1 root root  11110 Aug 20 04:47  facebook.png
-rw-r--r--. 1 root root    936 Aug 20 04:47  contact.html
-rw-r--r--. 1 root root  34395 Aug 20 04:47  buisnesscard.png
-rw-r--r--. 1 root root  86675 Aug 20 04:47  brochure.jpg
-rw-r--r--. 1 root root  69398 Aug 20 04:47  banner.png
-rw-r--r--. 1 root root  29997 Aug 20 04:47  audio.mp3
-rw-r--r--. 1 root root      2 Aug 20 04:47  about.html
[root@ip-172-31-51-115 html]#
```

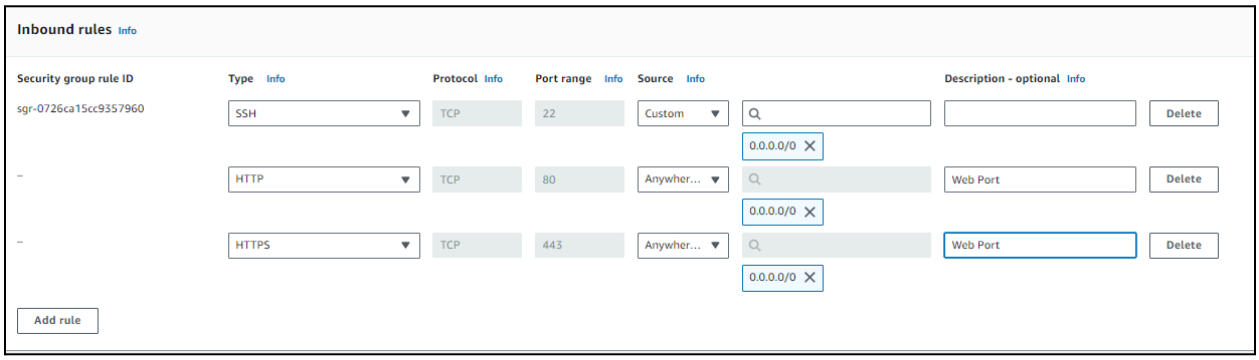
3) Editing the Inbound Rules:

Click on “Edit Inbound Rules” Button:





Now, Add new Rules as follows:



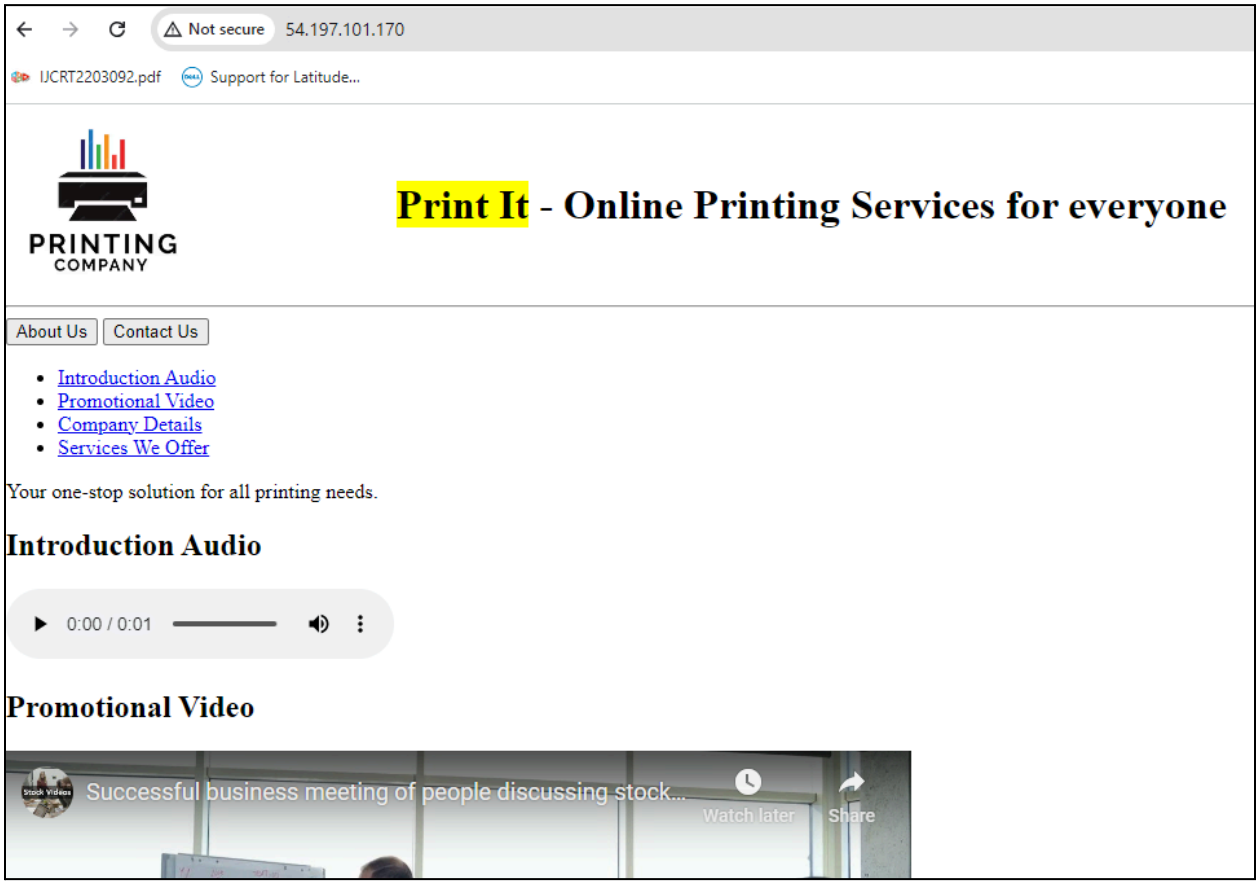
4) Check the status of httpd and then enable & start httpd using the following commands:

```
systemctl status httpd
systemctl enable httpd
systemctl start httpd
```

Now open the public ipv4 address allocated to the EC2 instance we created in new tab. We will be able to see the Website.

```
[root@ip-172-31-51-115 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-172-31-51-115 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service → /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-51-115 html]# start httpd
-bash: start: command not found
[root@ip-172-31-51-115 html]# systemctl start httpd
[root@ip-172-31-51-115 html]#
```

```
[root@ip-172-31-51-115 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 2024-08-21 20:59:52 UTC; 3s ago
       Docs: man:httpd.service(8)
   Main PID: 28176 (httpd)
     Status: "Started, listening on: port 80"
      Tasks: 177 (limit: 1112)
     Memory: 12.9M
        CPU: 63ms
     CGroup: /system.slice/httpd.service
             ├─28176 /usr/sbin/httpd -DFOREGROUND
             ├─28177 /usr/sbin/httpd -DFOREGROUND
             ├─28178 /usr/sbin/httpd -DFOREGROUND
             ├─28179 /usr/sbin/httpd -DFOREGROUND
             └─28180 /usr/sbin/httpd -DFOREGROUND

Aug 21 20:59:52 ip-172-31-51-115.ec2.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
Aug 21 20:59:52 ip-172-31-51-115.ec2.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
Aug 21 20:59:52 ip-172-31-51-115.ec2.internal httpd[28176]: Server configured, listening on: port 80
```



Not secure 54.197.101.170

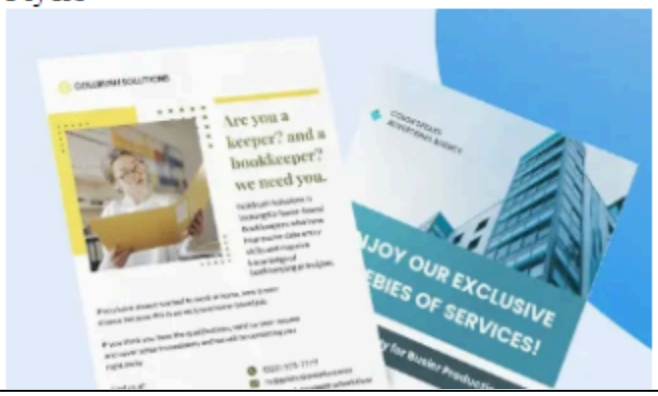
IJCRT2203092.pdf Support for Latitude...

Company Details

Services We Offer

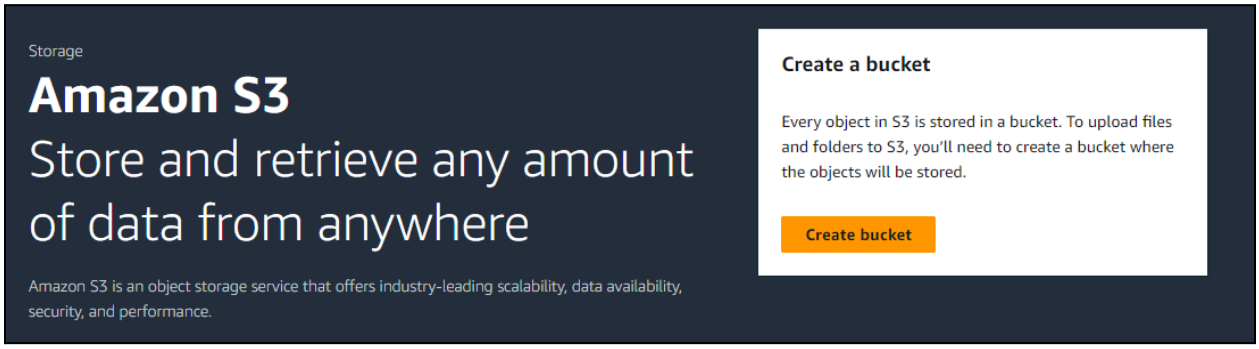
- Business Cards

- Flyers

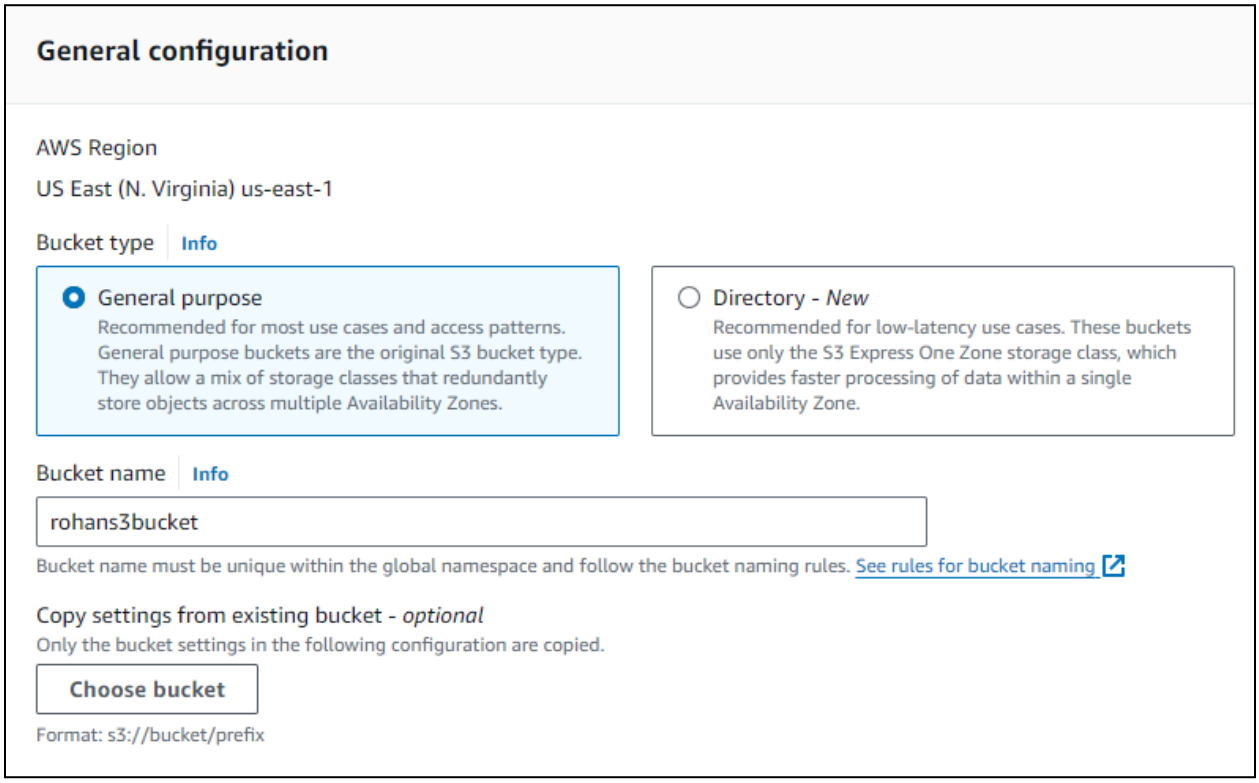


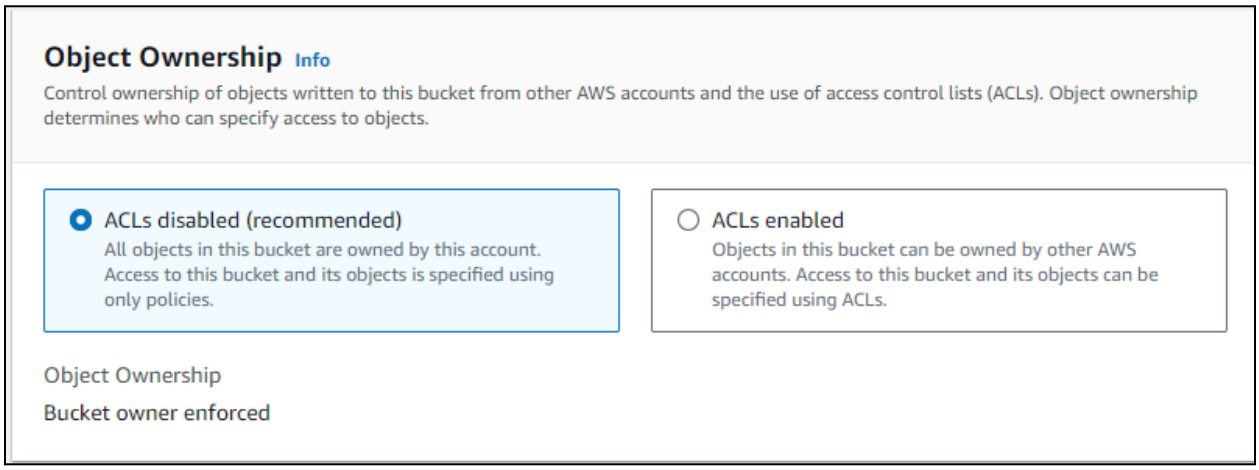
3) Using S3 Bucket:

1) Search for s3 bucket and go to the link and click on create bucket.



2) Configure the settings:





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

- ○ Disable
- ◉ Enable

Default encryption Info

Server-side encryption is automatically applied to new objects stored in this bucket.

Encryption type Info

- ◉ Server-side encryption with Amazon S3 managed keys (SSE-S3)
- ○ Server-side encryption with AWS Key Management Service keys (SSE-KMS)
- ○ Dual-layer server-side encryption with AWS Key Management Service keys (DSSE-KMS)

  Secure your objects with two separate layers of encryption. For details on pricing, see **DSSE-KMS pricing** on the **Storage** tab of the Amazon S3 pricing page.

Bucket Key

Using an S3 Bucket Key for SSE-KMS reduces encryption costs by lowering calls to AWS KMS. S3 Bucket Keys aren't supported for DSSE-KMS. Learn more

- ○ Disable
- ◉ Enable

General purpose buckets (1) Info All AWS Regions

Buckets are containers for data stored in S3.

Copy ARN | Empty | Delete | Create bucket

Find buckets by name

1

Name	AWS Region	IAM Access Analyzer	Creation date
rohans3bucket1	US East (N. Virginia) us-east-1	View analyzer for us-east-1	August 23, 2024, 21:32:27 (UTC+05:30)

3) Upload the files.

Files and folders (13 Total, 456.9 KB)

Remove | Add files | Add folder

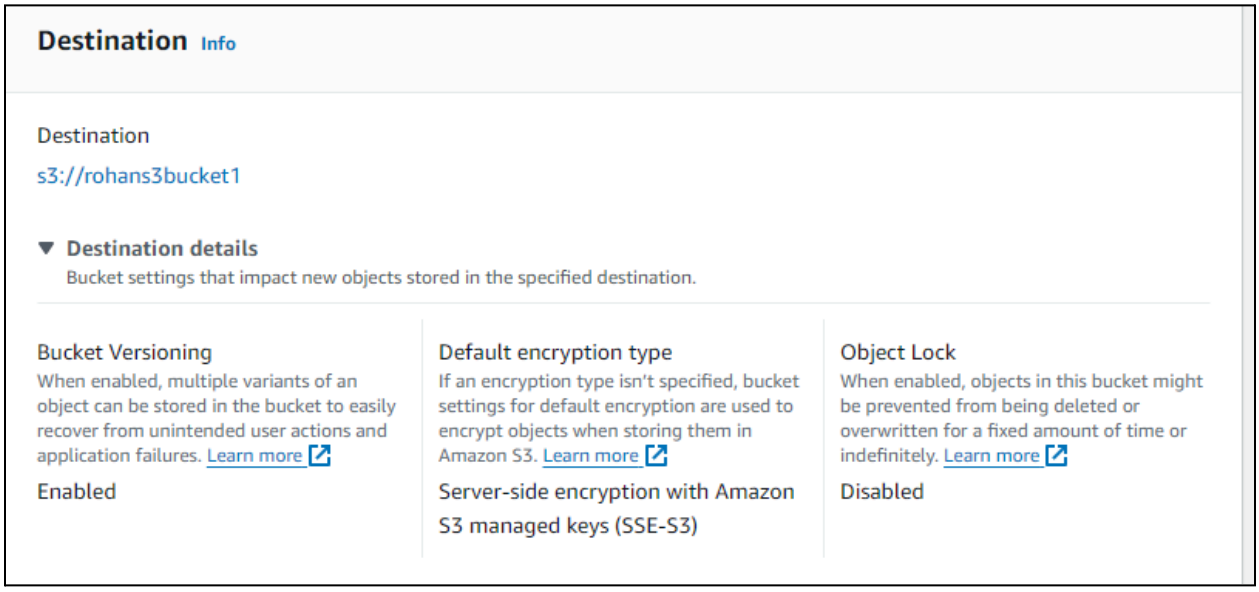
All files and folders in this table will be uploaded.

Find by name

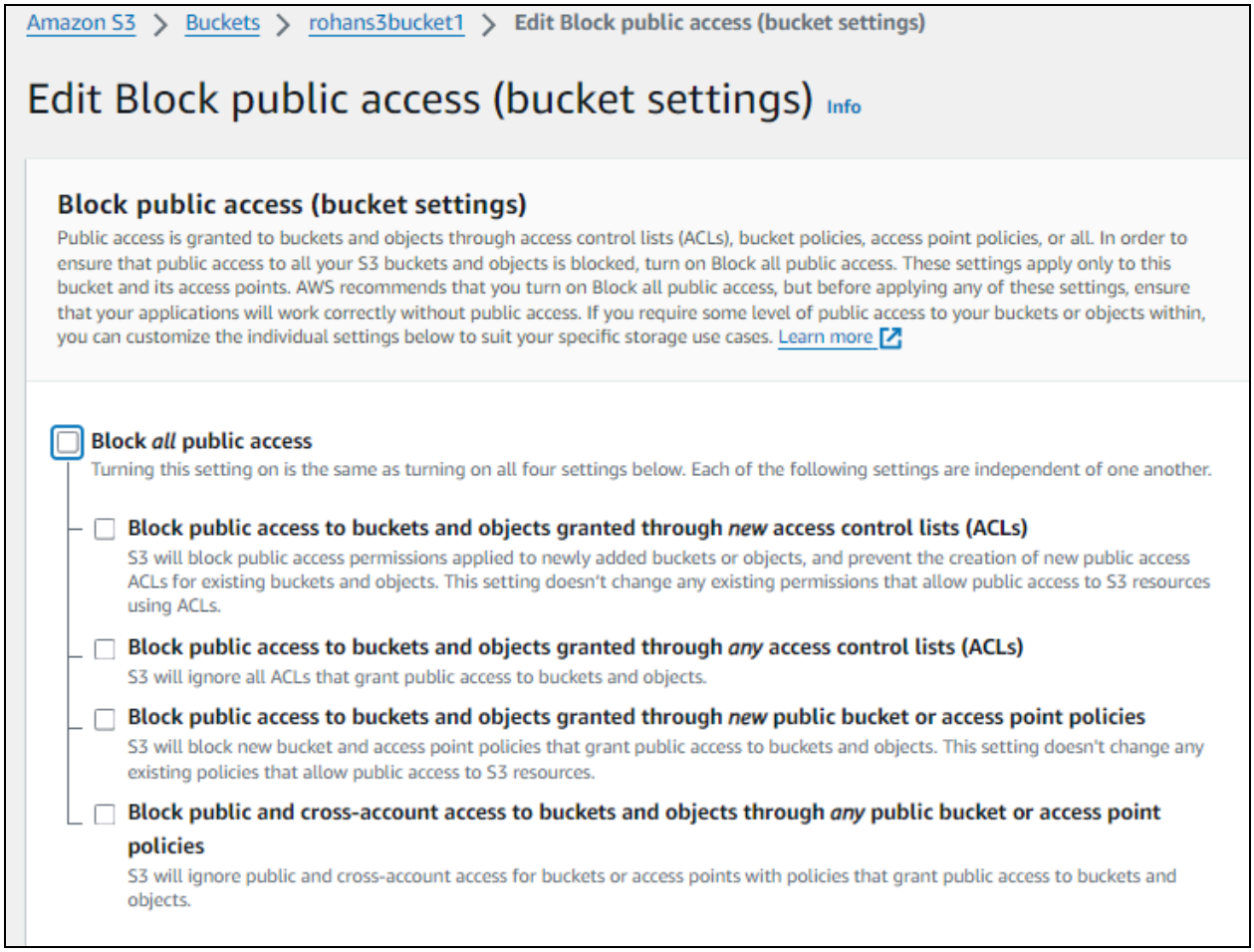
1 2

Name	Folder	Type
x.png	-	image/png
about.html	-	text/html
audio.mp3	-	audio/mpeg
banner.png	-	image/png
brochure.jpg	-	image/jpeg
buisnesscard.png	-	image/png
contact.html	-	text/html
facebook.png	-	image/png
flyers.png	-	image/png
index.html	-	text/html

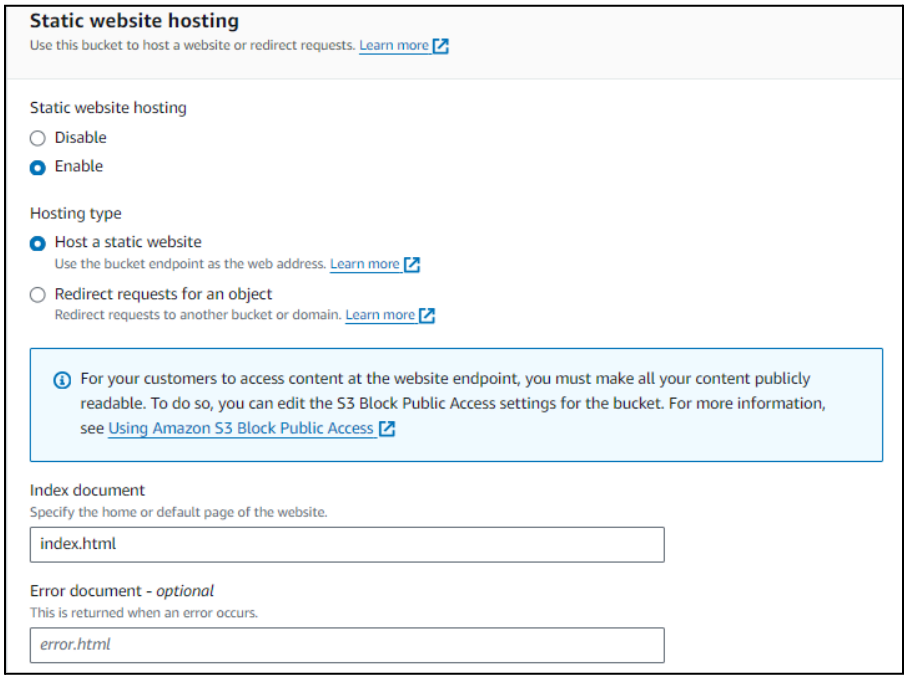
4) Go to your bucket destination link.



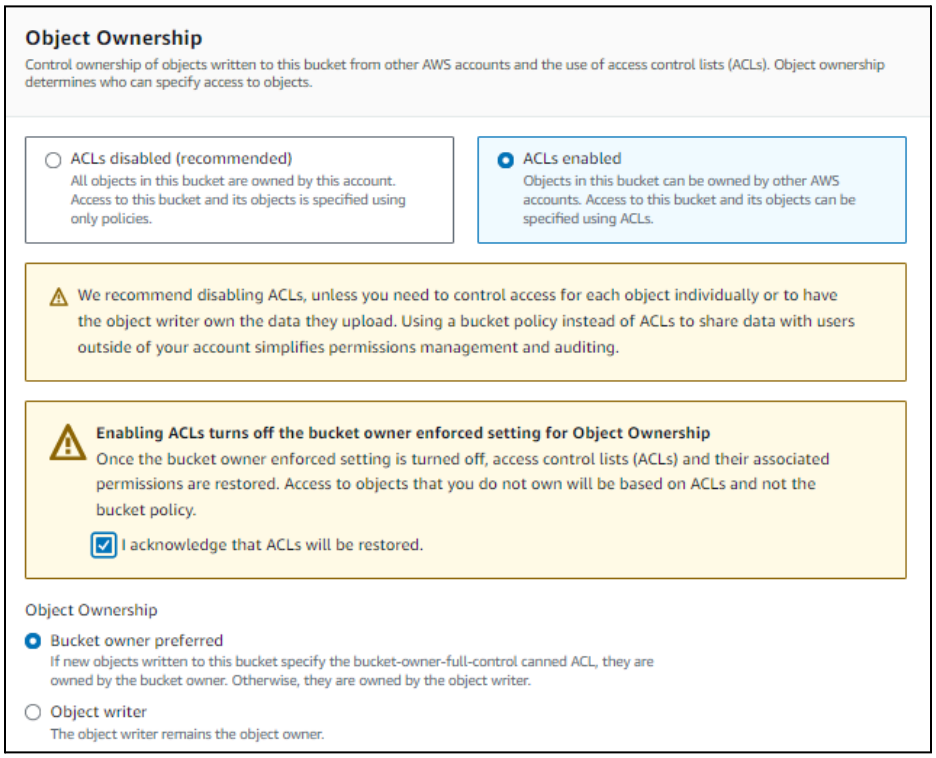
5) Uncheck the block public access.



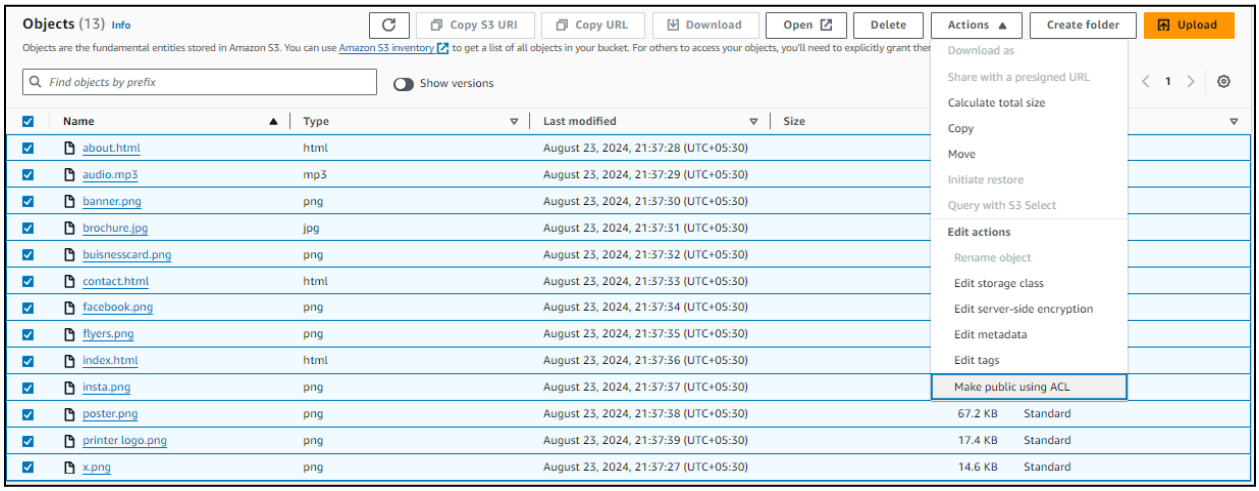
6) Go to the properties section and scroll down to static website hosting and enable it.



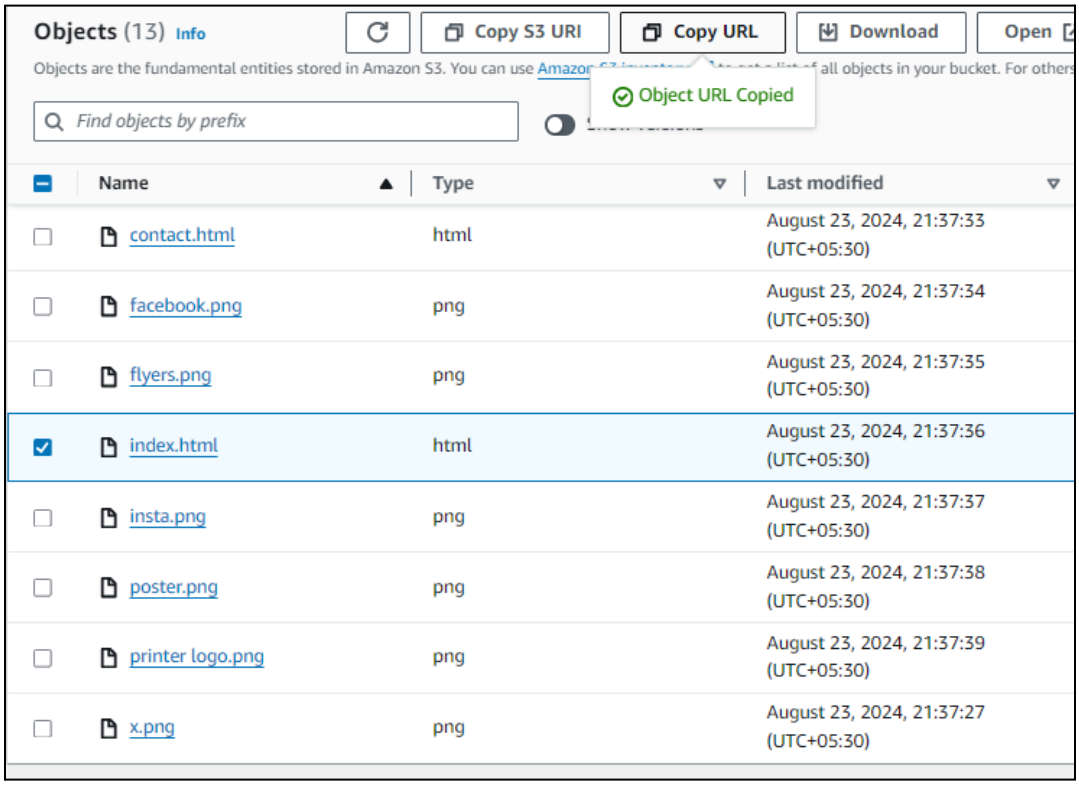
7) Go to the permissions section and see for ACL's option and enable it.



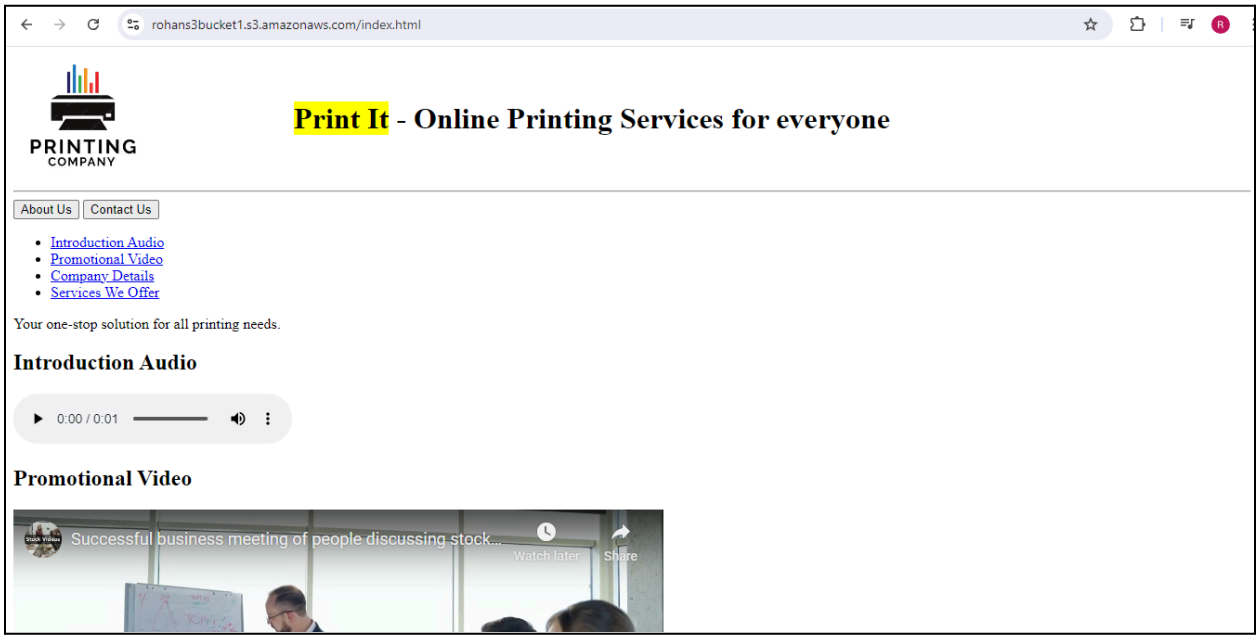
8) Now, go to the objects section and select all objects and click on action button and select “Make public using ACL”



9) Copy the URL of the index.html file and go to the copied link in the new tab of the browser.

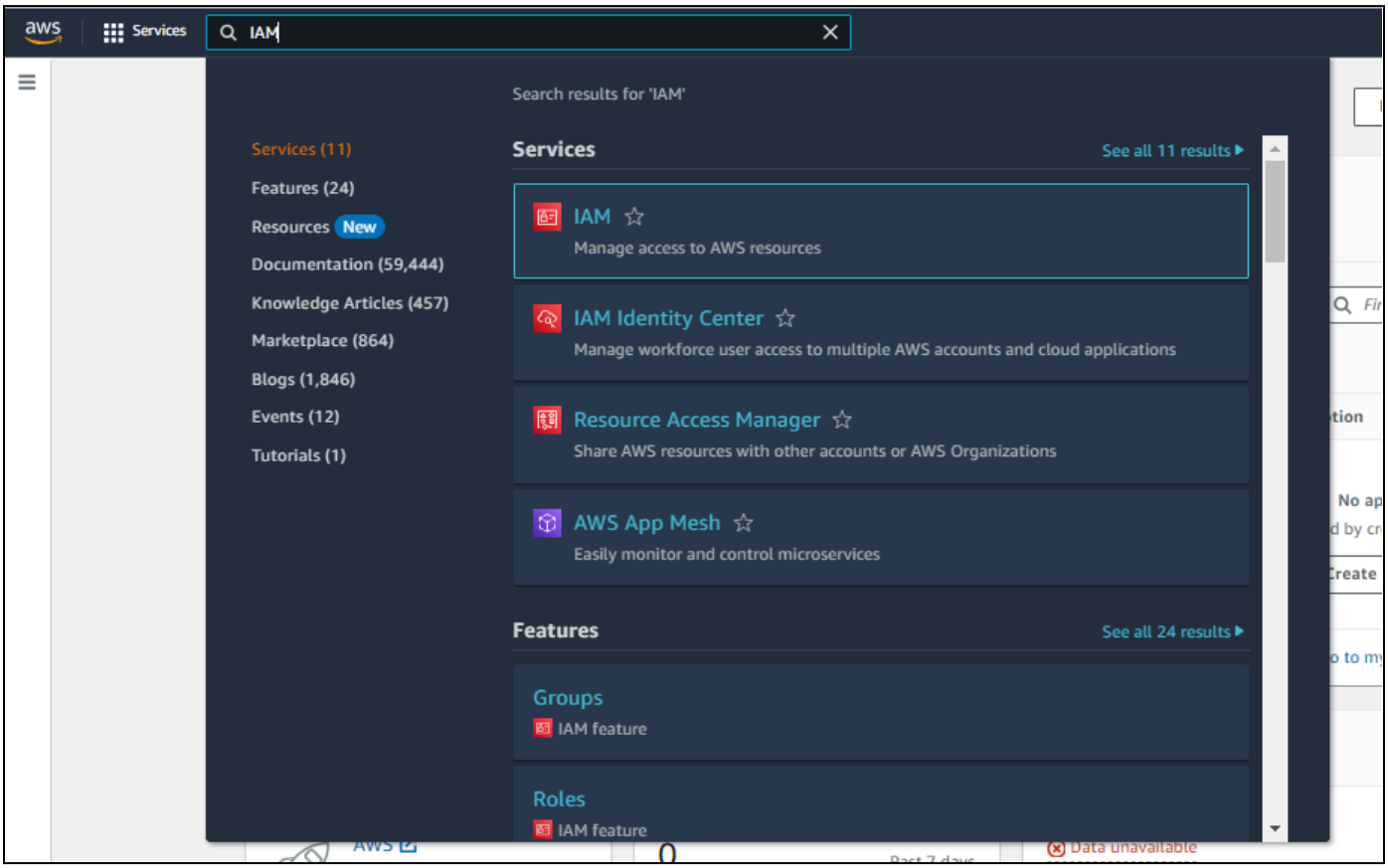


10) Website is successfully deployed.

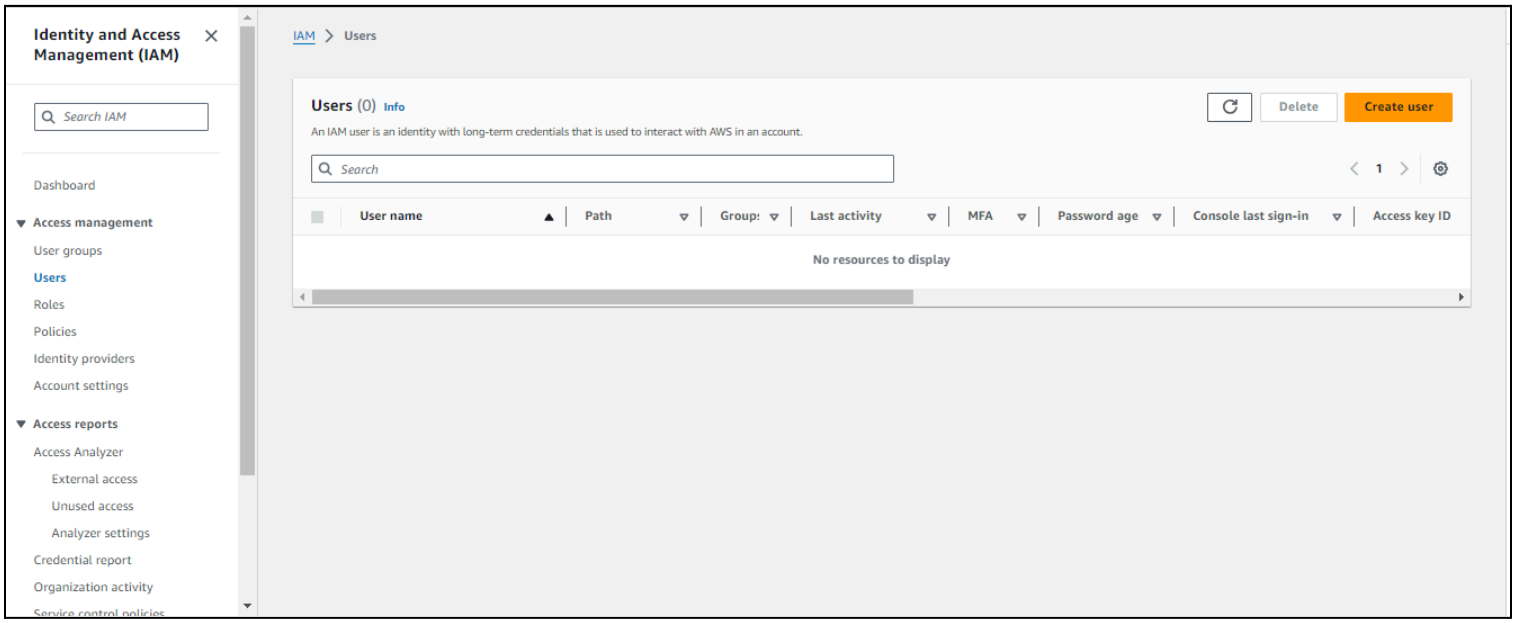


Part B) AWS Cloud9:

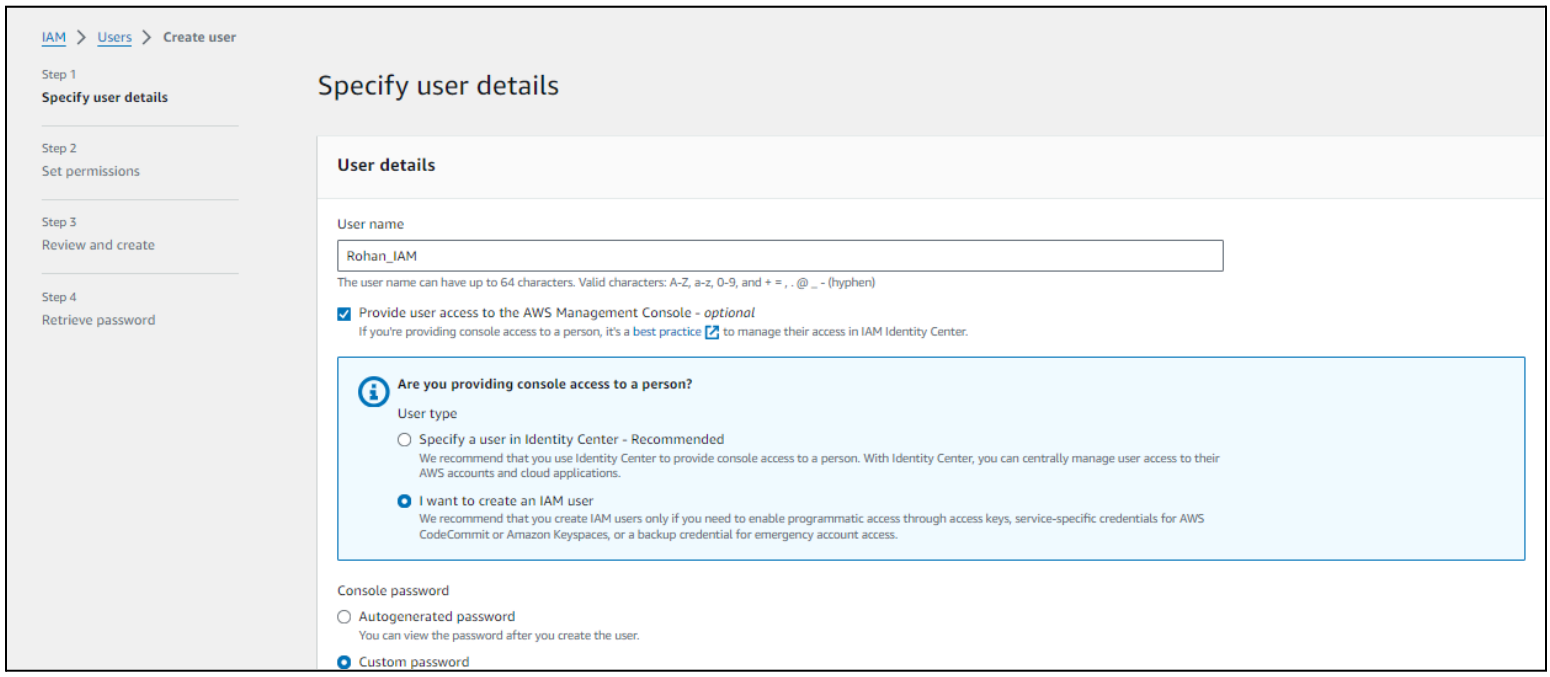
1) Search for IAM in the search box.

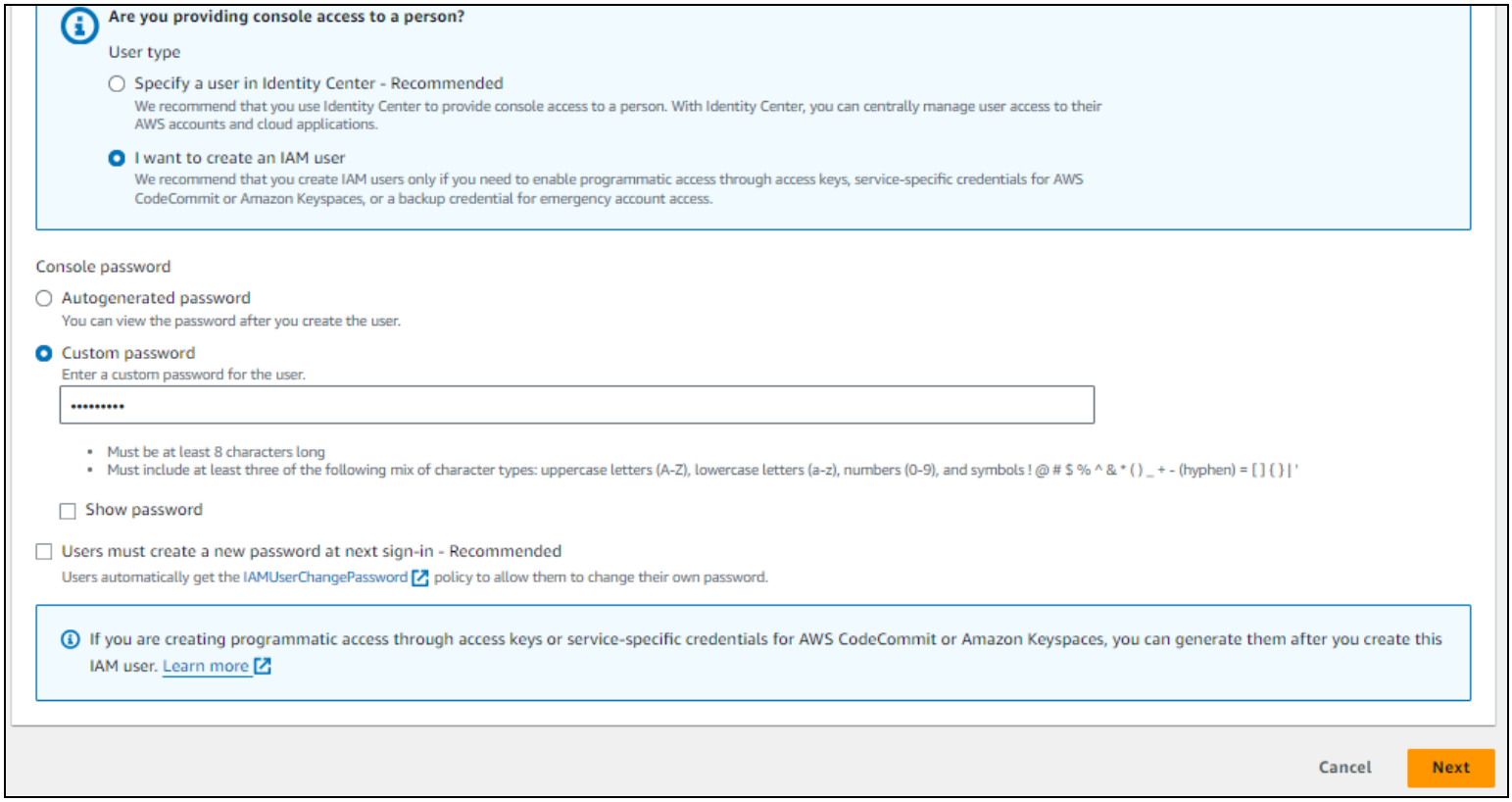


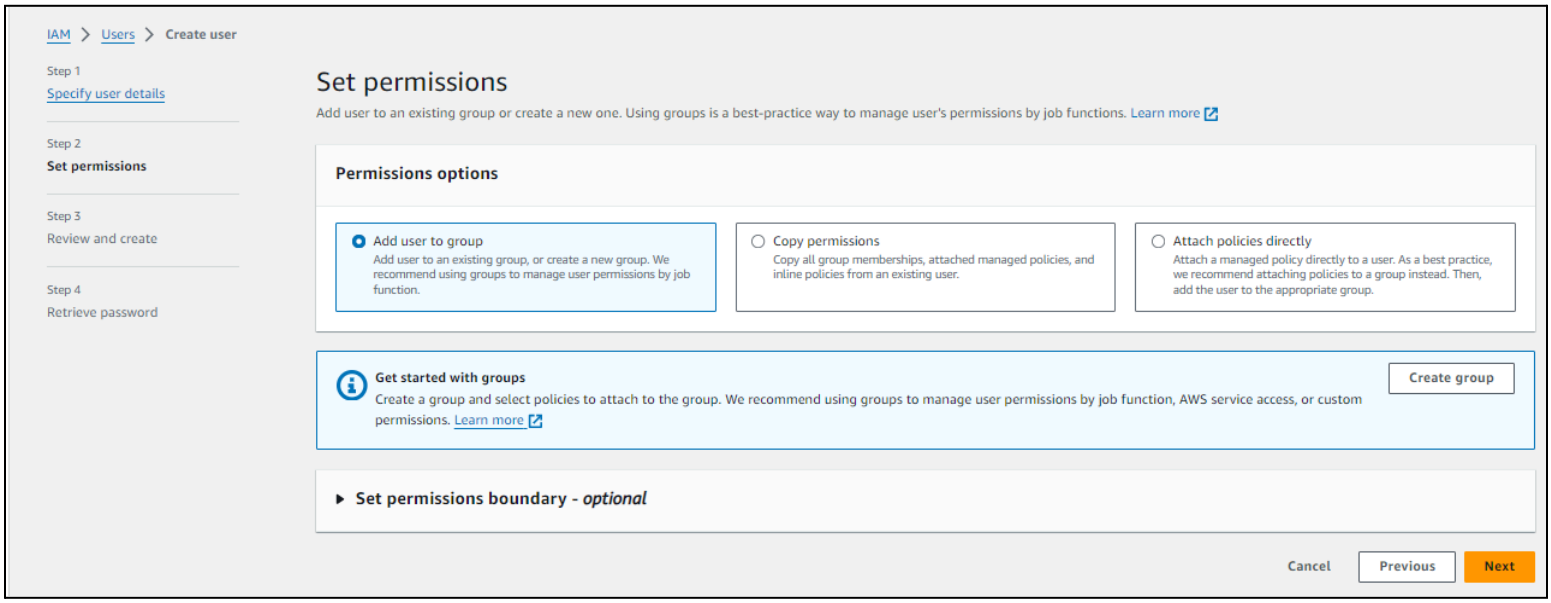
2) Click on create user.

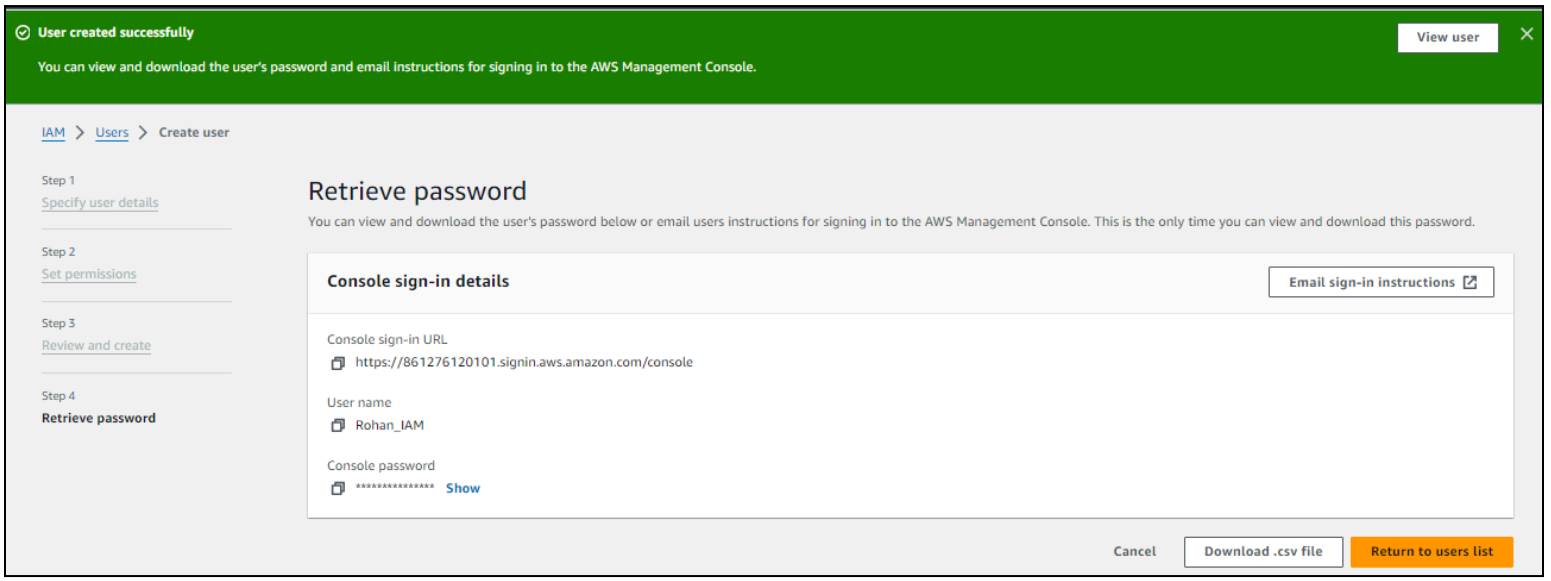


3) Configuring the IAM role.

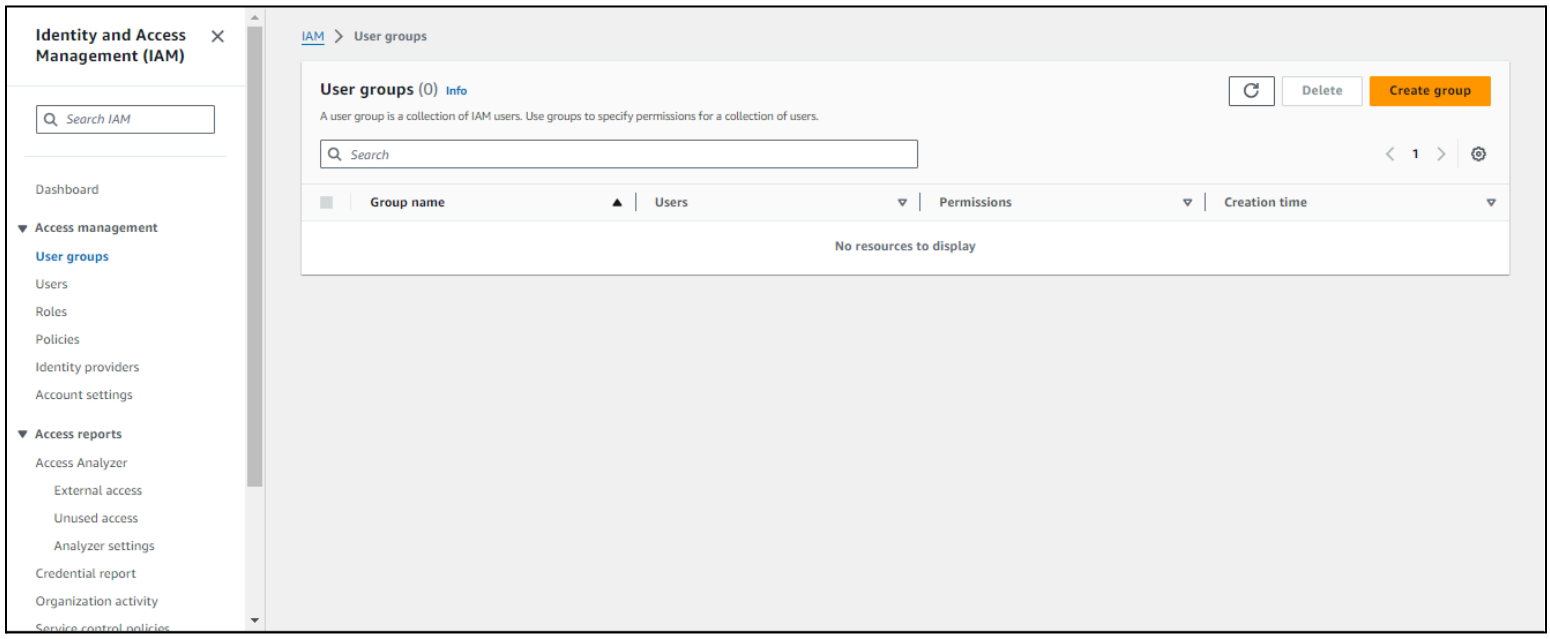




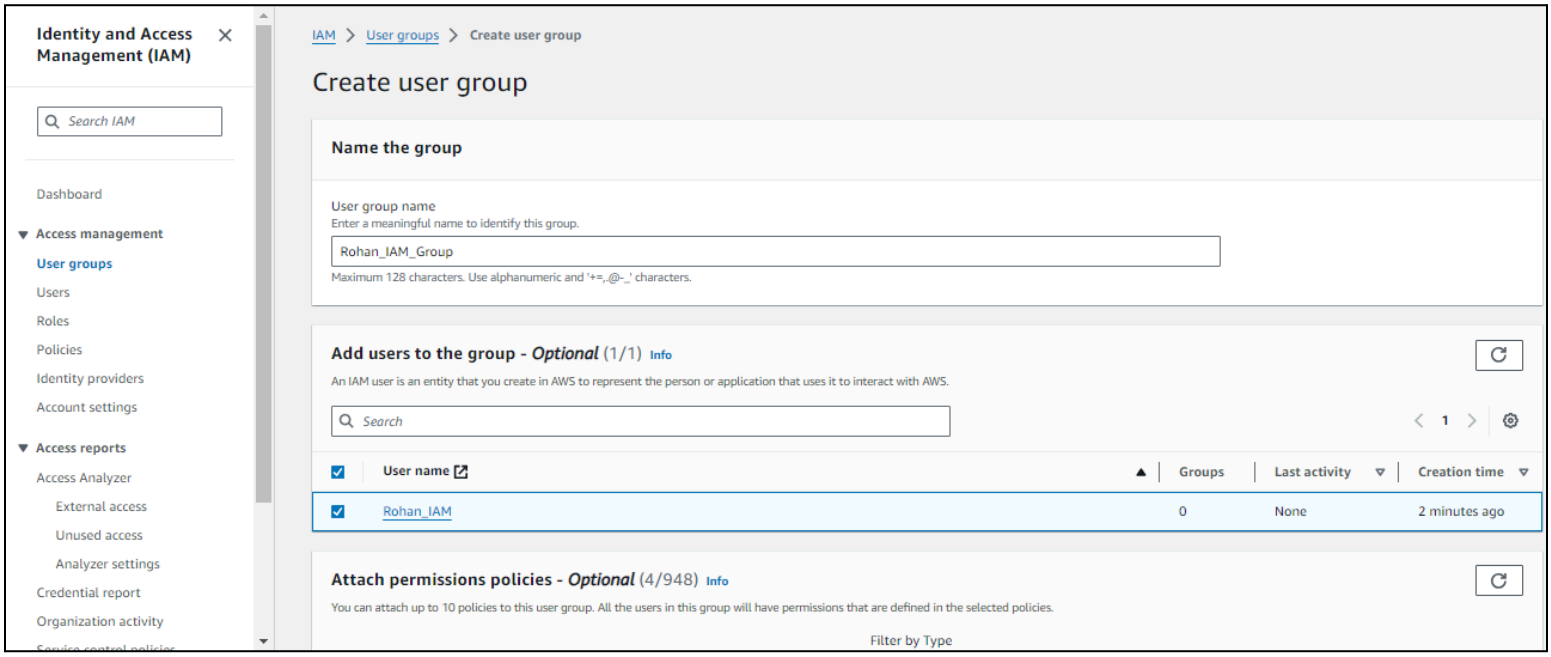

IAM > Users > Create user
Step 1
Specify user details
Step 2
Set permissions
Step 3
Review and create
Step 4
Retrieve password
Set permissions
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Permissions options
Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions by job function.
Copy permissions
Copy all group memberships, attached managed policies, and inline policies from an existing user.
Attach policies directly
Attach a managed policy directly to a user. As a best practice, we recommend attaching policies to a group instead. Then, add the user to the appropriate group.
Get started with groups
Create a group and select policies to attach to the group. We recommend using groups to manage user permissions by job function, AWS service access, or custom permissions. Learn more
Create group
Set permissions boundary - optional
Cancel
Previous
Next

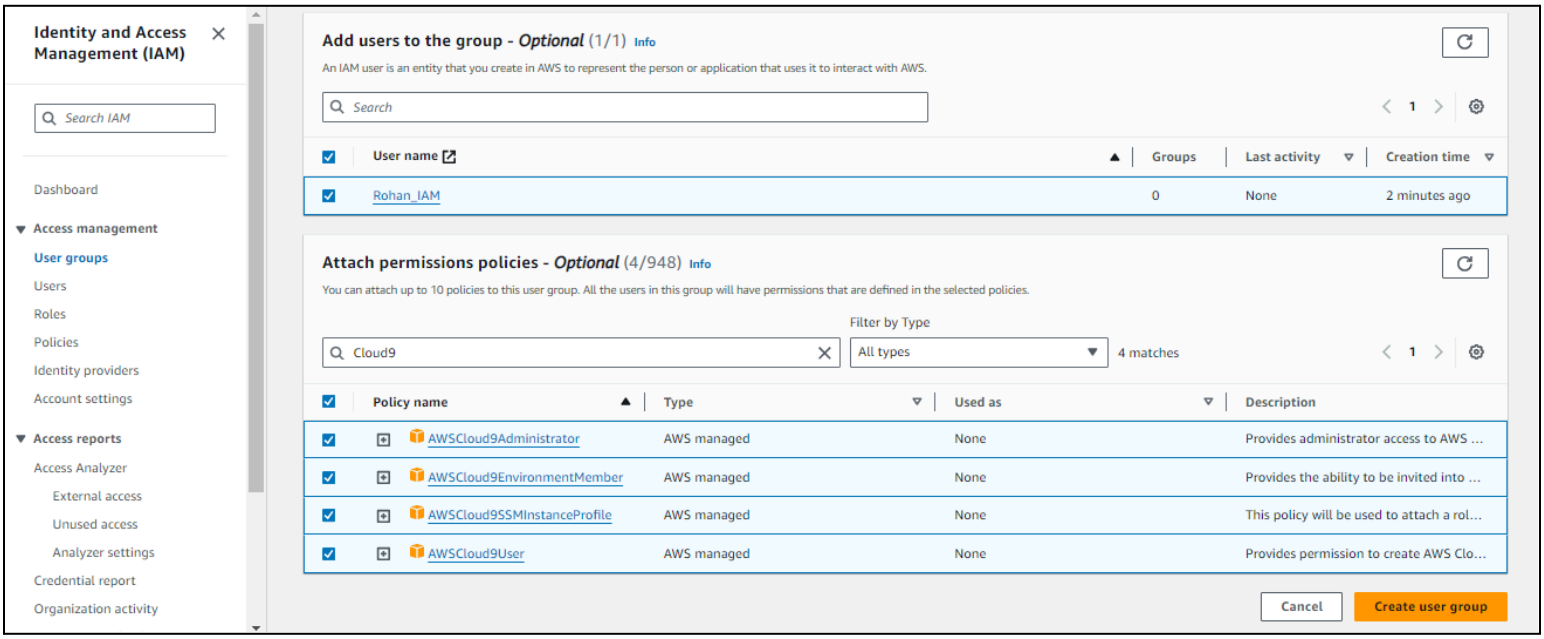

User created successfully
You can view and download the user's password and email instructions for signing in to the AWS Management Console.
View user
IAM > Users > Create user
Step 1
Specify user details
Step 2
Set permissions
Step 3
Review and create
Step 4
Retrieve password
Retrieve password
You can view and download the user's password below or email users instructions for signing in to the AWS Management Console. This is the only time you can view and download this password.
Console sign-in details
Email sign-in instructions
Console sign-in URL
https://861276120101.signin.aws.amazon.com/console
User name
Rohan_IAM
Console password
********** Show
Cancel
Download .csv file
Return to users list

4) After creating IAM role, go to User groups, and click on create group.

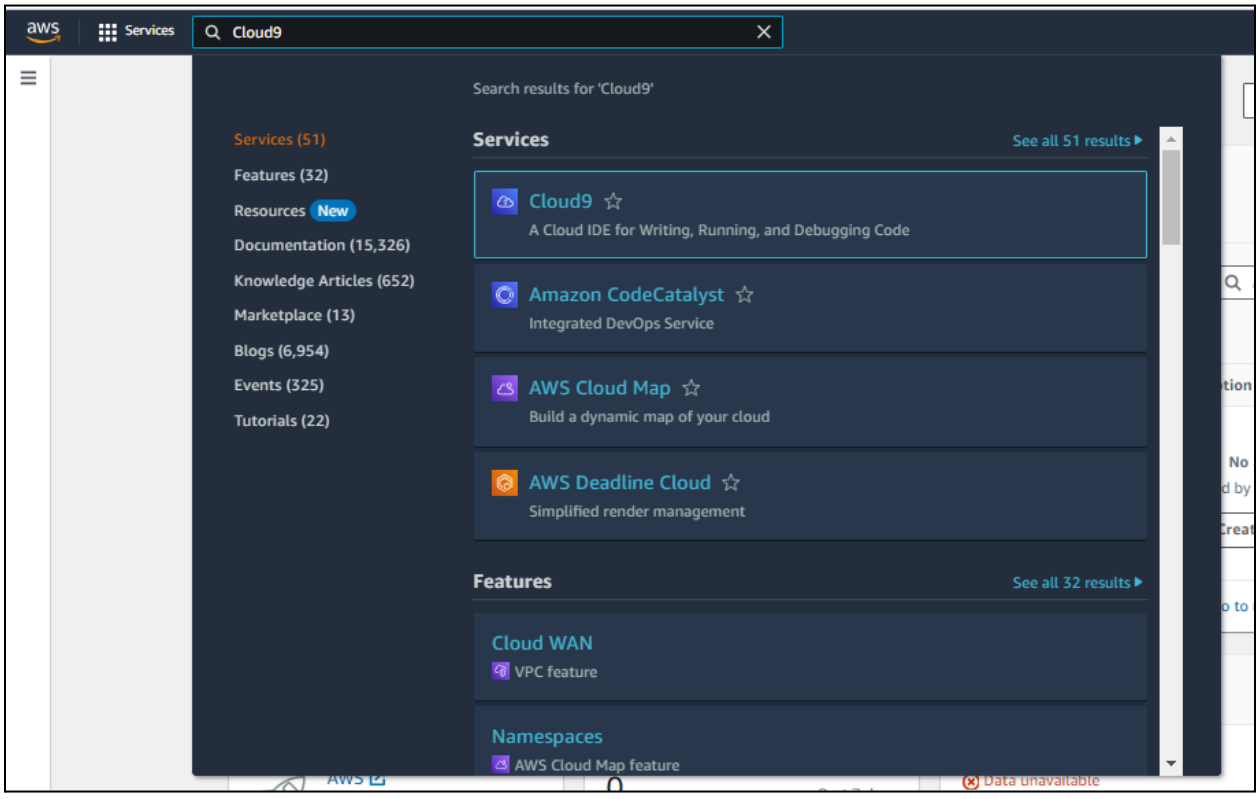


5) Configuring the user group.

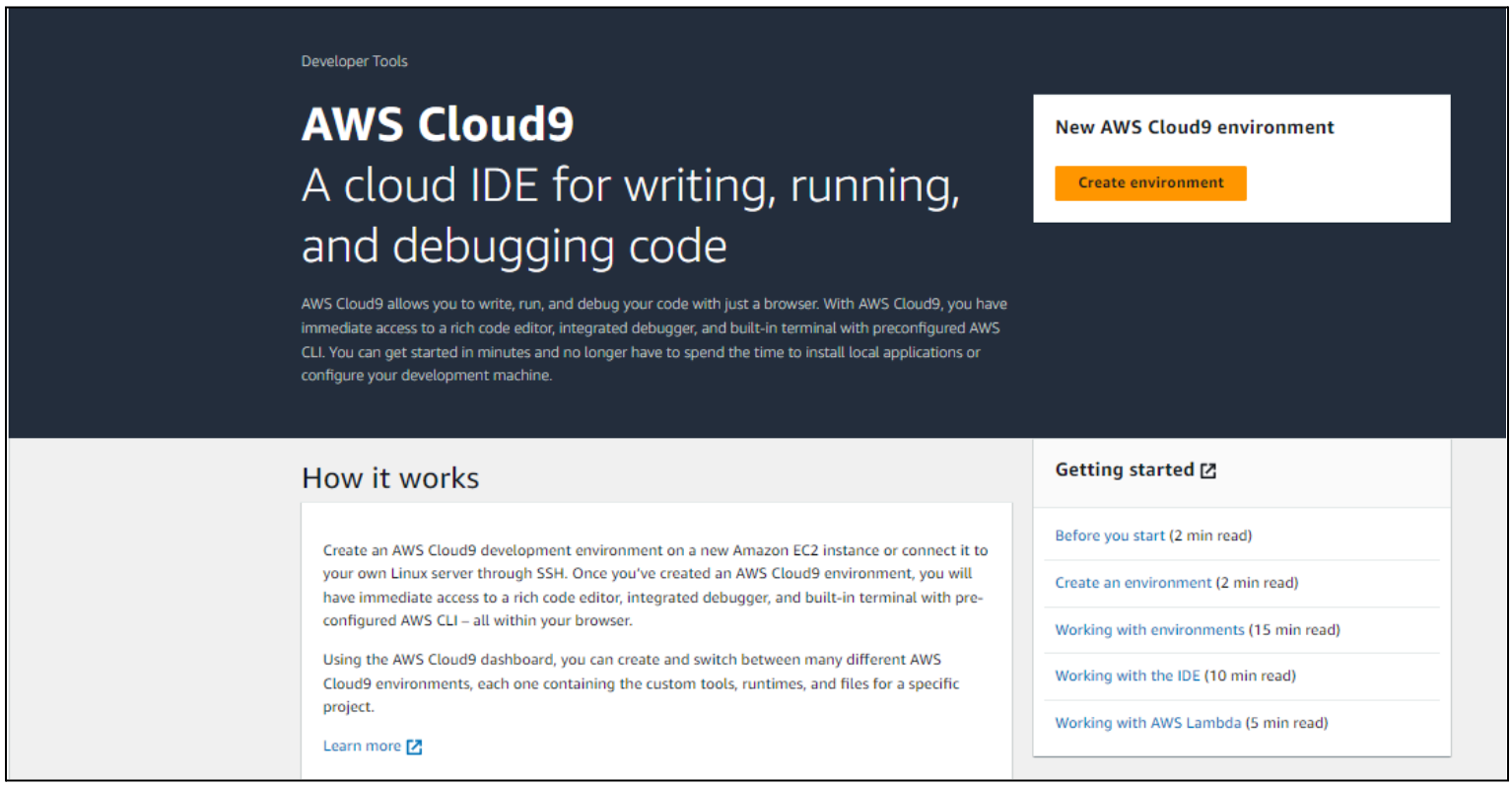




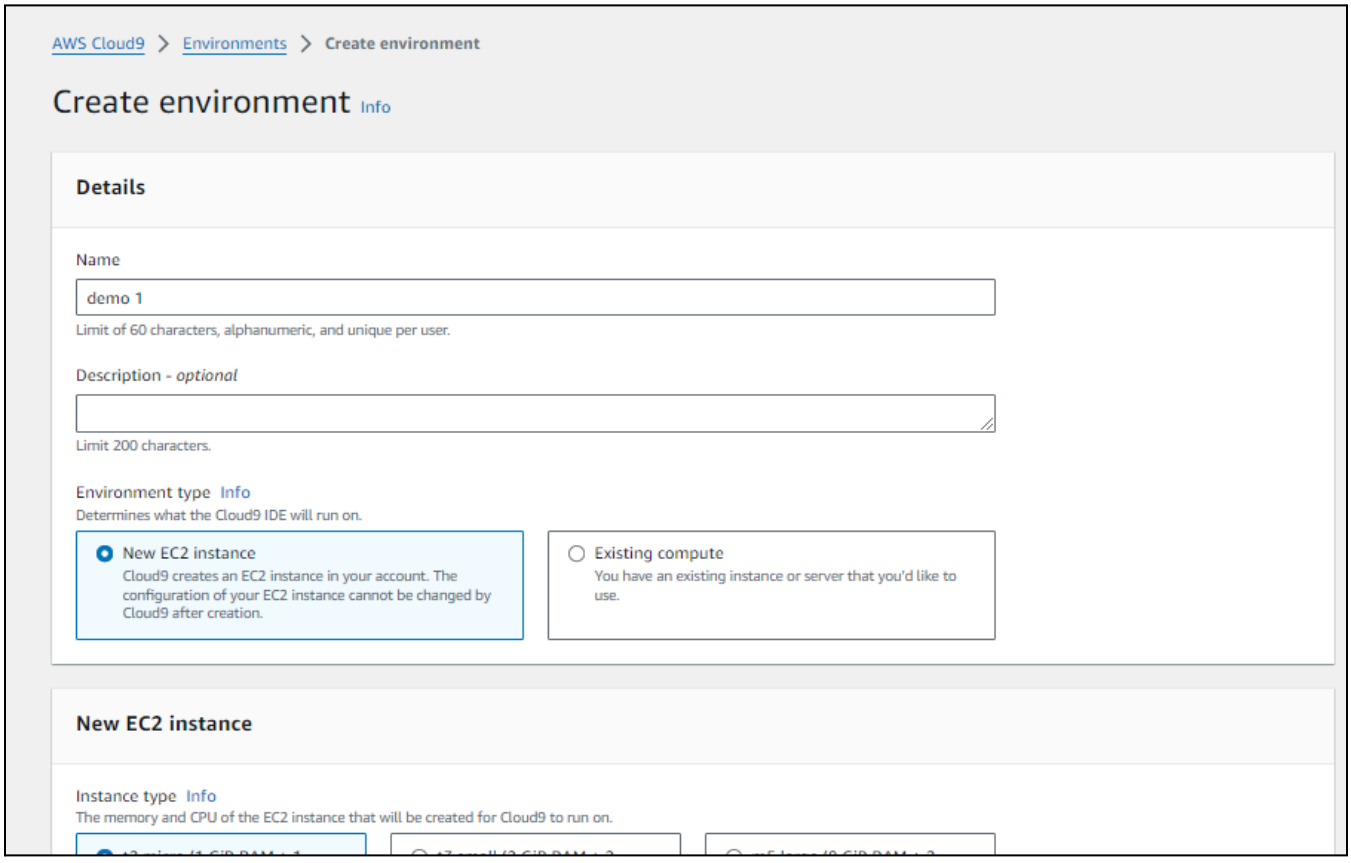
6) After creating a user group, search for Cloud9 in the search box.

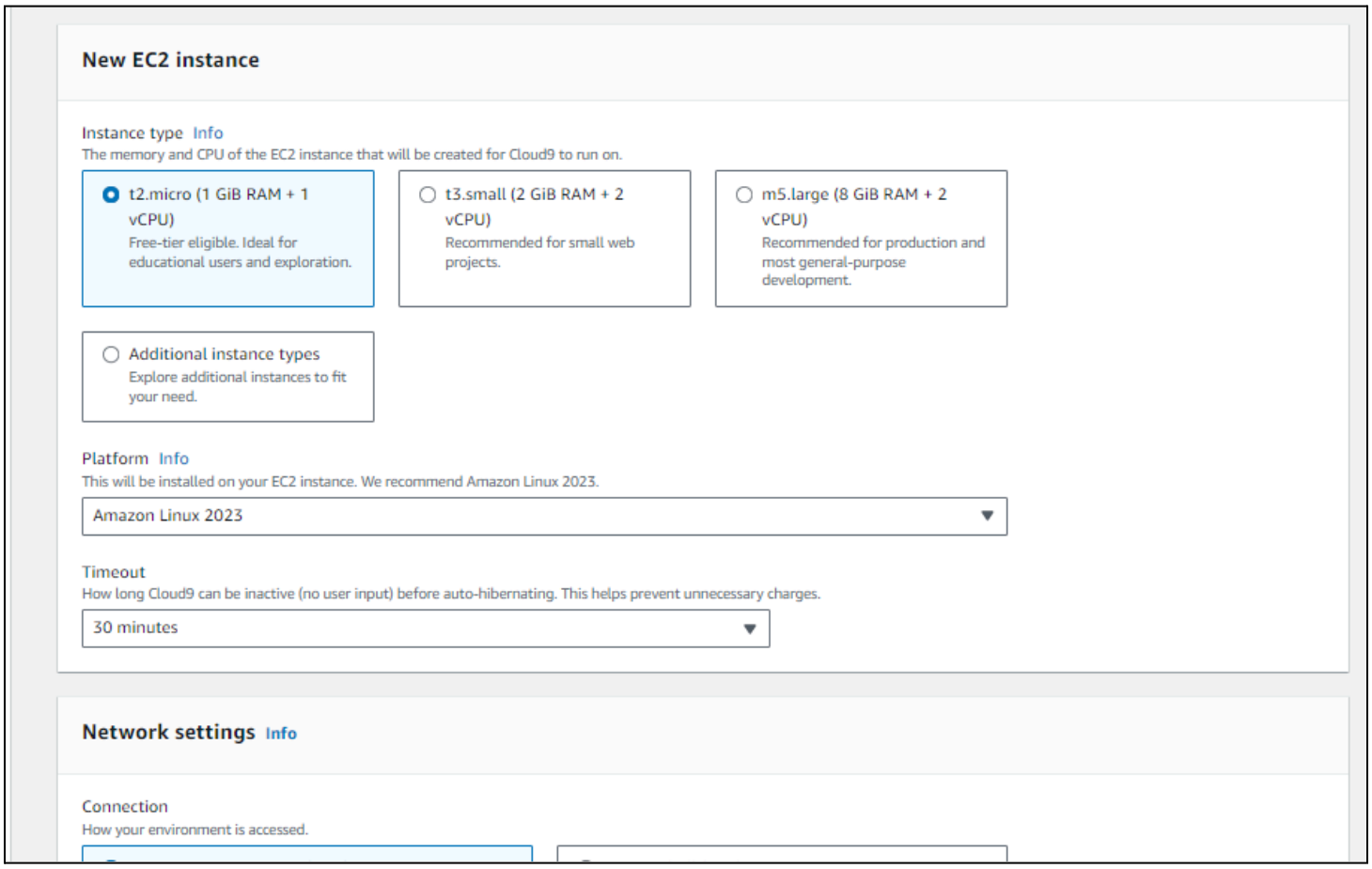


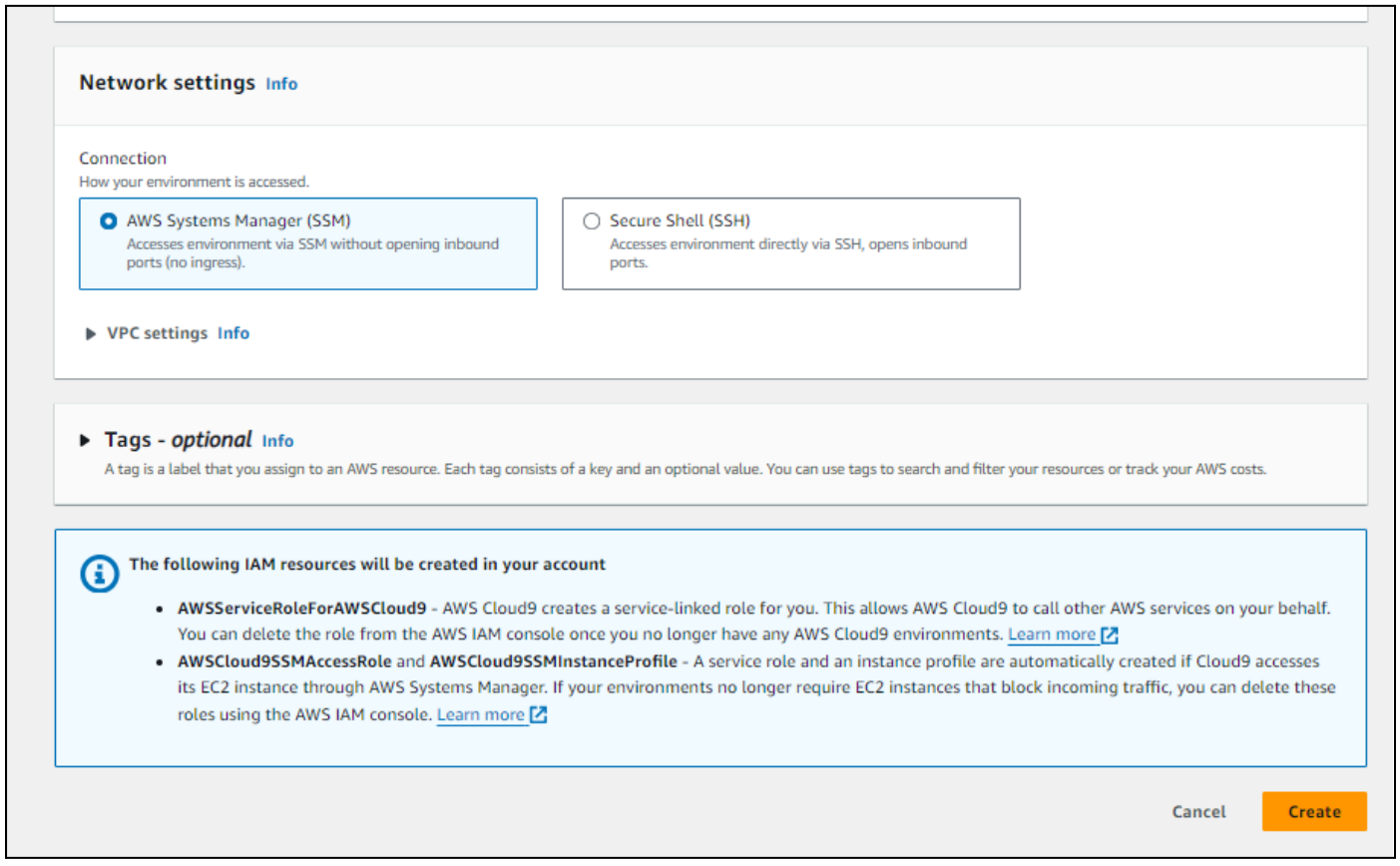
7) Click on create environment.



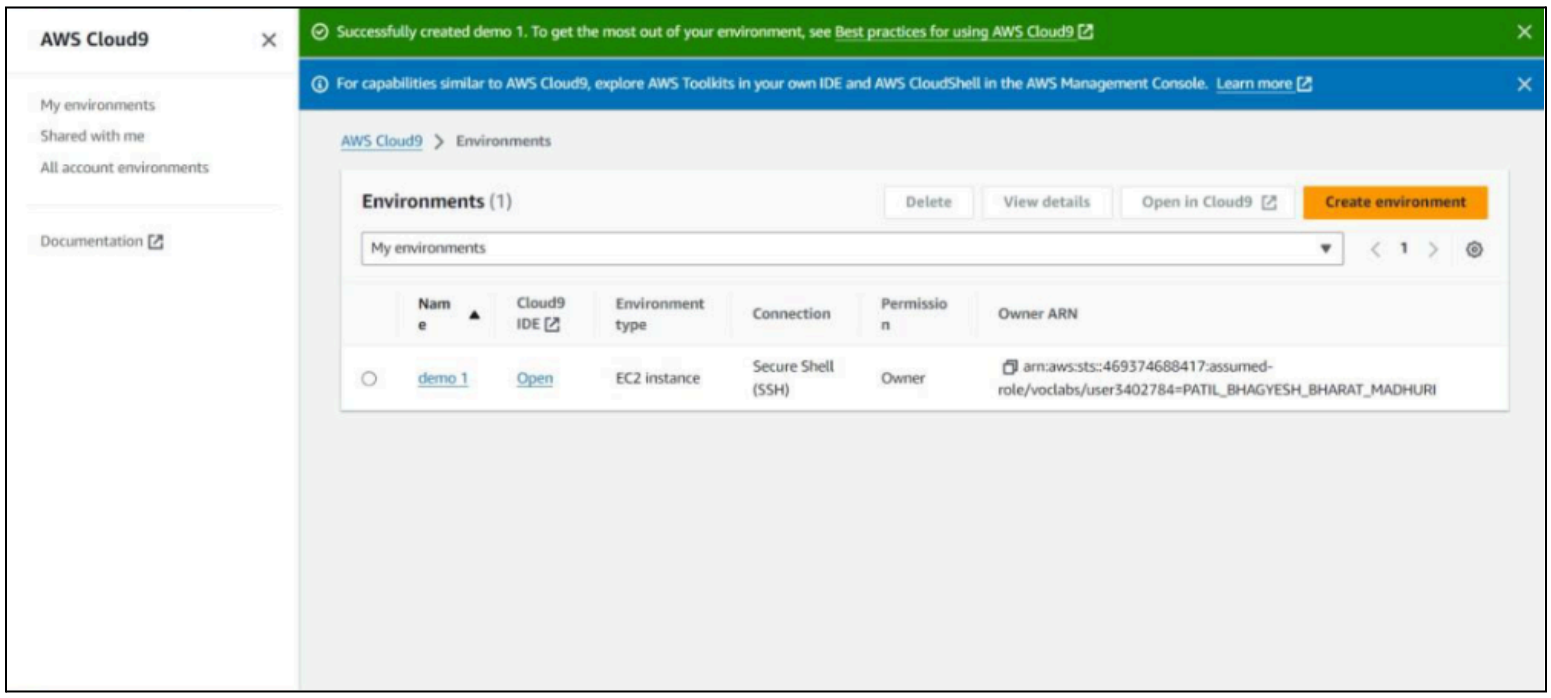
8) Configuring the environment.

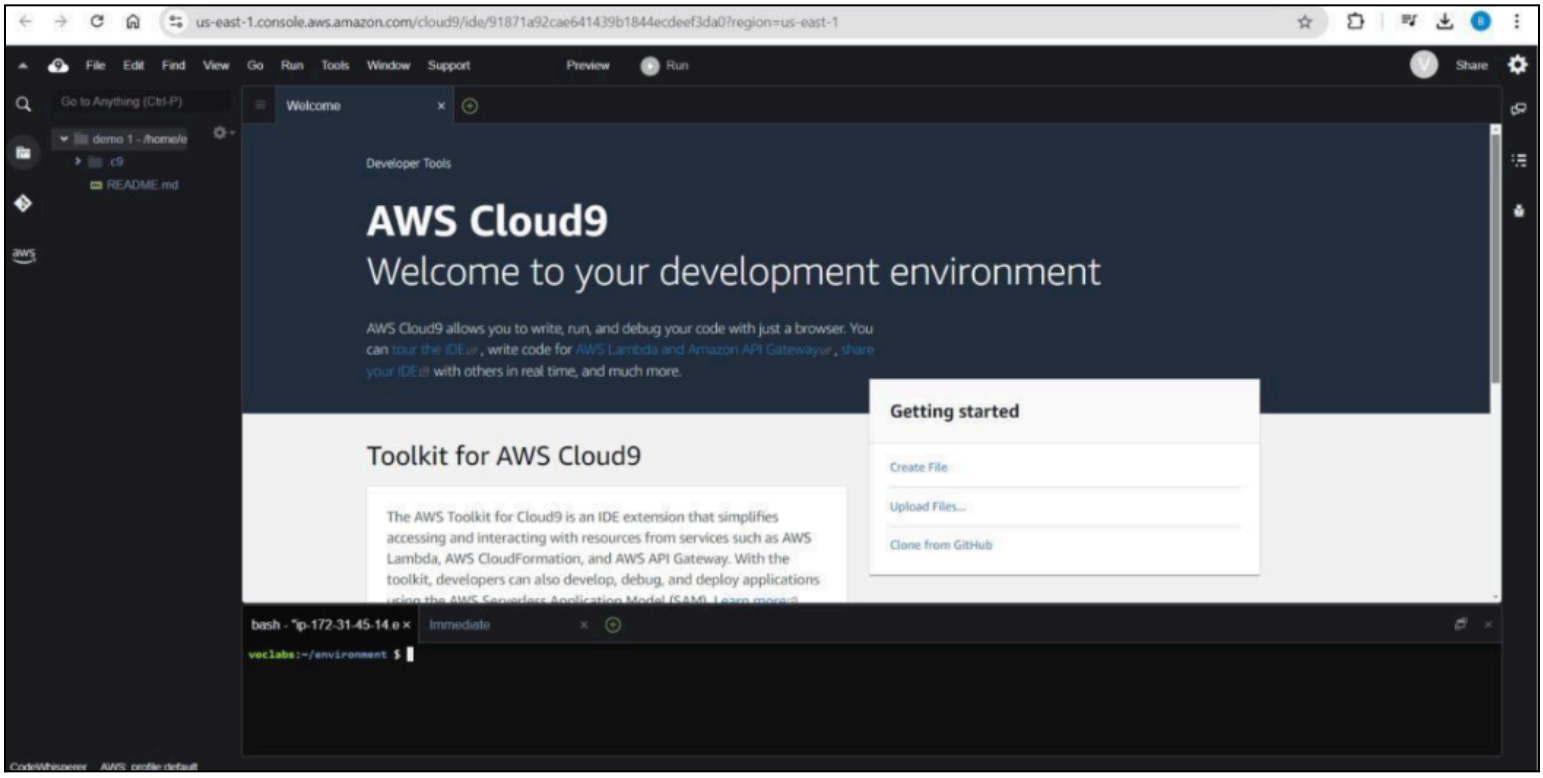


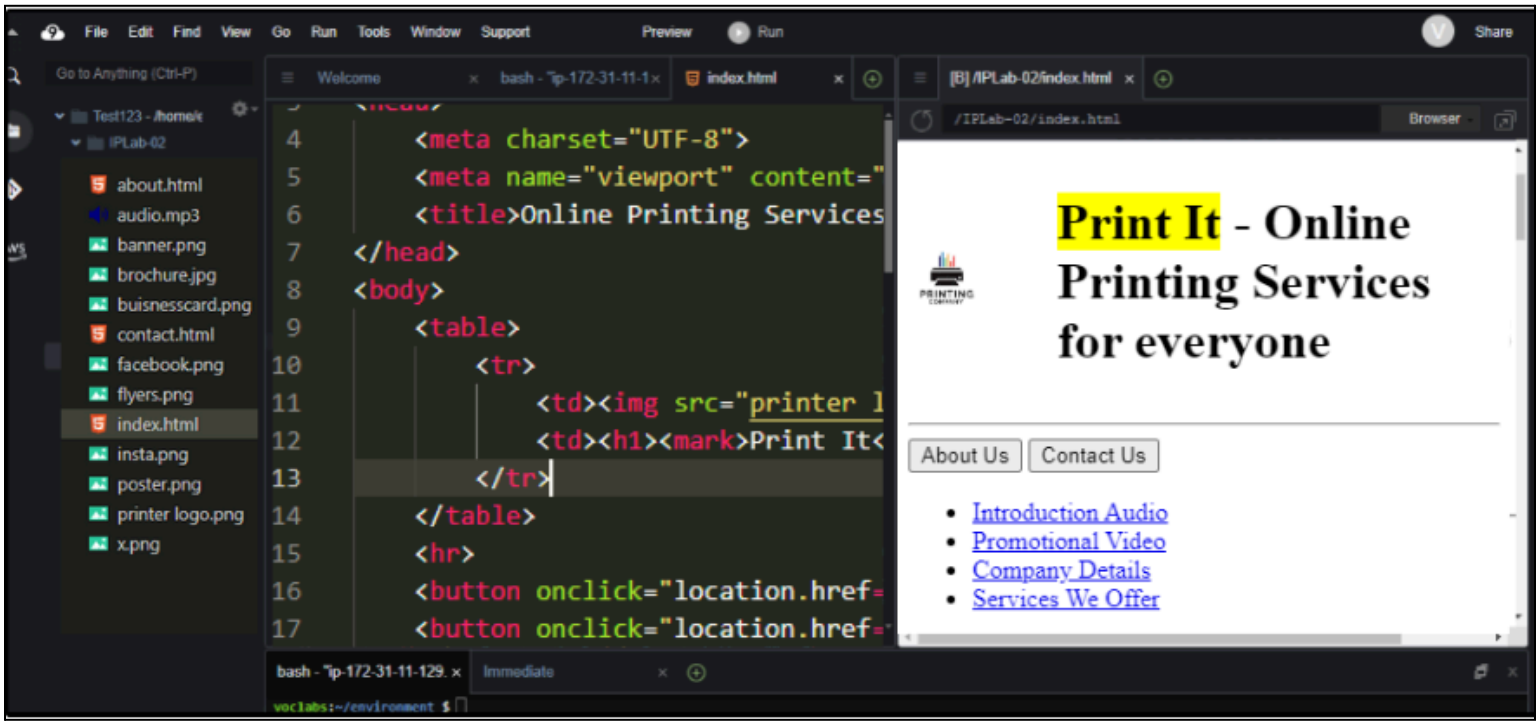

New EC2 instance
Instance type Info
The memory and CPU of the EC2 instance that will be created for Cloud9 to run on.
t2.micro (1 GiB RAM + 1 vCPU)
Free-tier eligible. Ideal for educational users and exploration.
t3.small (2 GiB RAM + 2 vCPU)
Recommended for small web projects.
m5.large (8 GiB RAM + 2 vCPU)
Recommended for production and most general-purpose development.
Additional instance types
Explore additional instances to fit your need.
Platform Info
This will be installed on your EC2 instance. We recommend Amazon Linux 2023.
Amazon Linux 2023
Timeout
How long Cloud9 can be inactive (no user input) before auto-hibernating. This helps prevent unnecessary charges.
30 minutes
Network settings Info
Connection
How your environment is accessed.


Network settings Info
Connection
How your environment is accessed.
AWS Systems Manager (SSM)
Accesses environment via SSM without opening inbound ports (no ingress).
Secure Shell (SSH)
Accesses environment directly via SSH, opens inbound ports.
VPC settings Info
Tags - optional Info
A tag is a label that you assign to an AWS resource. Each tag consists of a key and an optional value. You can use tags to search and filter your resources or track your AWS costs.
The following IAM resources will be created in your account
AWSServiceRoleForAWSCloud9 - AWS Cloud9 creates a service-linked role for you. This allows AWS Cloud9 to call other AWS services on your behalf. You can delete the role from the AWS IAM console once you no longer have any AWS Cloud9 environments. Learn more
AWSCloud9SSMAccessRole and AWSCloud9SSMInstanceProfile - A service role and an instance profile are automatically created if Cloud9 accesses its EC2 instance through AWS Systems Manager. If your environments no longer require EC2 instances that block incoming traffic, you can delete these roles using the AWS IAM console. Learn more
Cancel
Create

9) Environment is successfully created.






File Edit Find View Go Run Tools Window Support Preview Run Share
Go to Anything (Ctrl-P)
Test123 - /home/e
IPLab-02
about.html
audio.mp3
banner.png
brochure.jpg
buisnesscard.png
contact.html
facebook.png
flyers.png
index.html
insta.png
poster.png
printer logo.png
x.png
Welcome
bash - "ip-172-31-11-1
index.html
<meta charset="UTF-8">
<meta name="viewport" content="
<title>Online Printing Services
</head>
<body>
<table>
<tr>
<td><img src="printer l
<td><h1><mark>Print It<
</tr>
</table>
<hr>
<button onclick="location.href=
<button onclick="location.href=
bash - "ip-172-31-11-129.
Immediate
voclabs:~/environment $
[B] /IPLab-02/index.html
/IPLab-02/index.html
Browser
Print It - Online Printing Services for everyone
About Us
Contact Us
Introduction Audio
Promotional Video
Company Details
Services We Offer