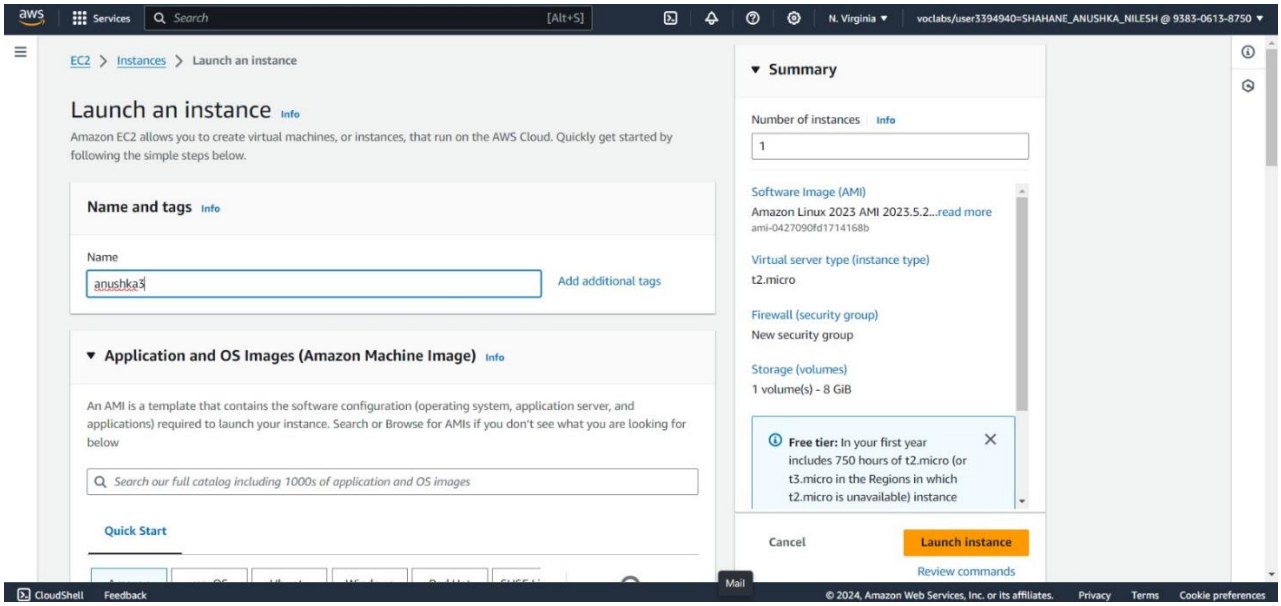
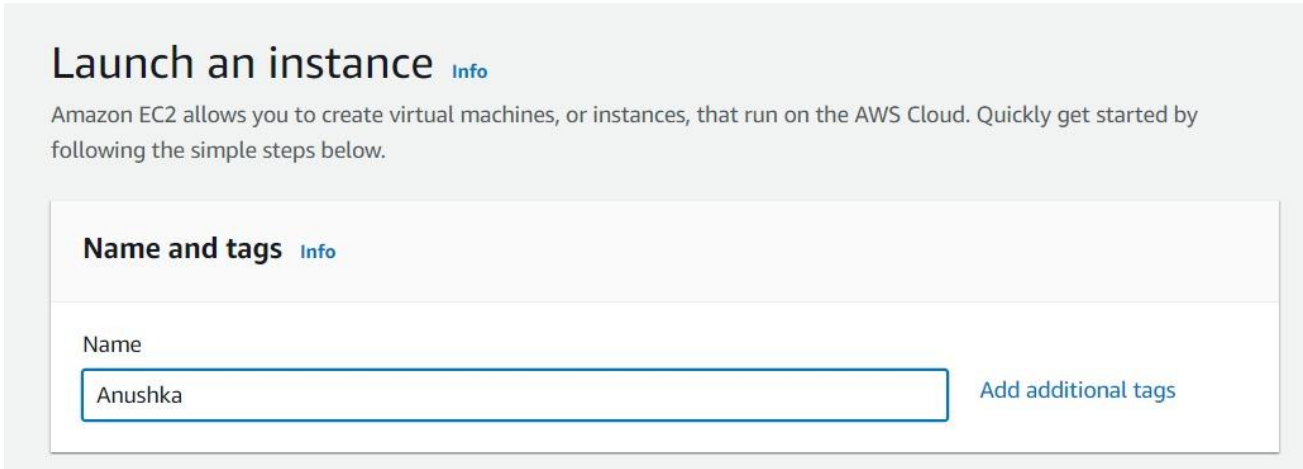
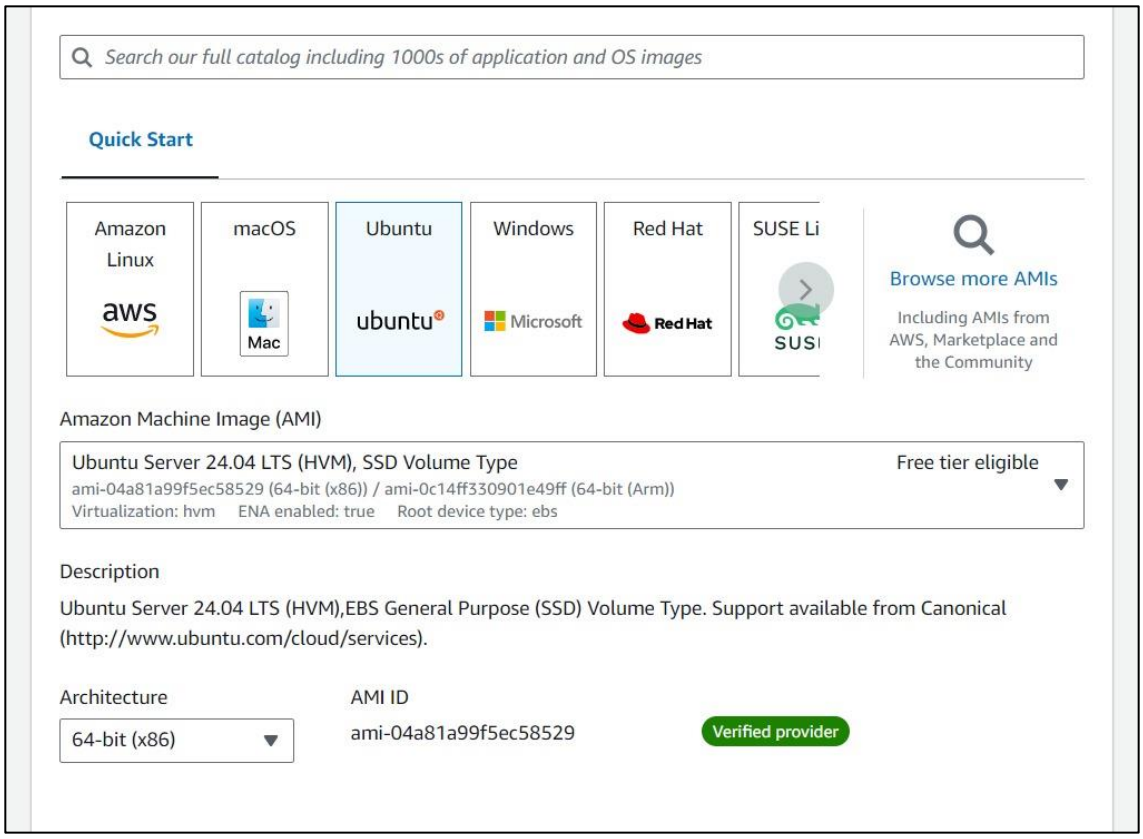
ADVANCE DEVOPS EXPERIMENT 1

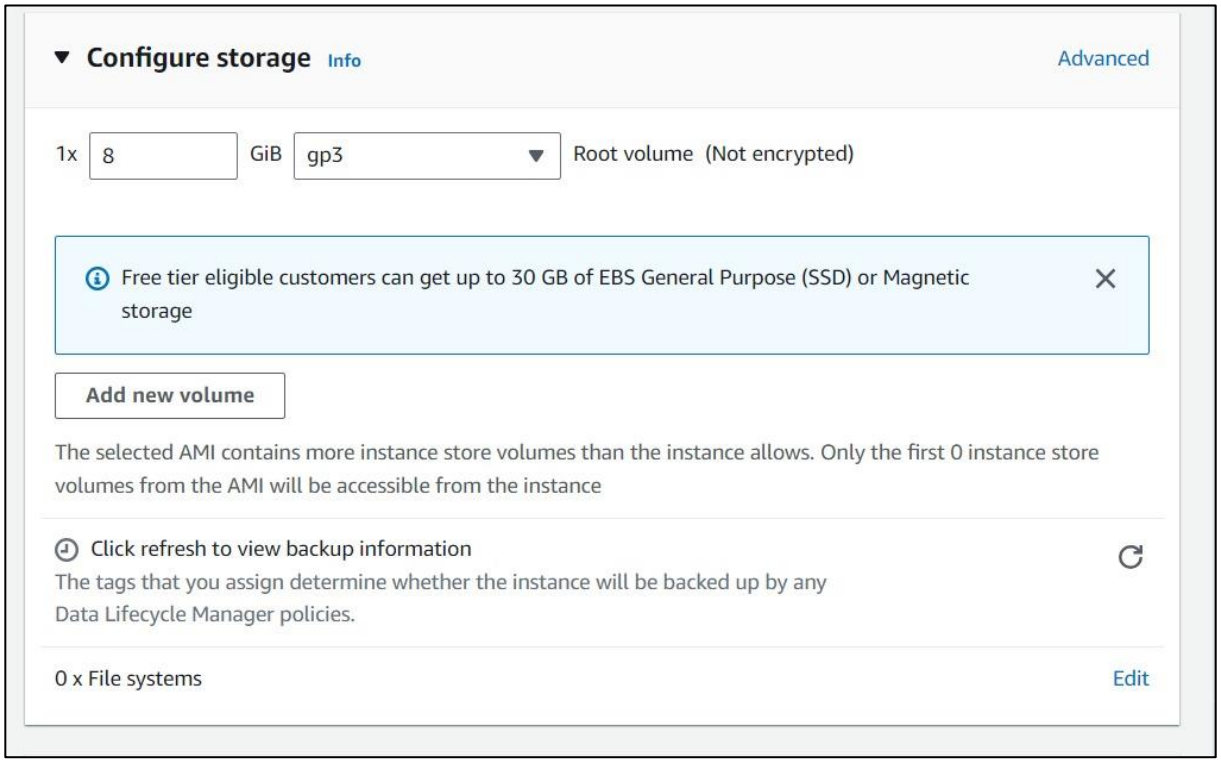
Aim : To understand the benefits of Cloud Infrastructure and Setup AWS Cloud9 IDE, Launch AWS Cloud9 IDE and Perform Collaboration Demonstration.



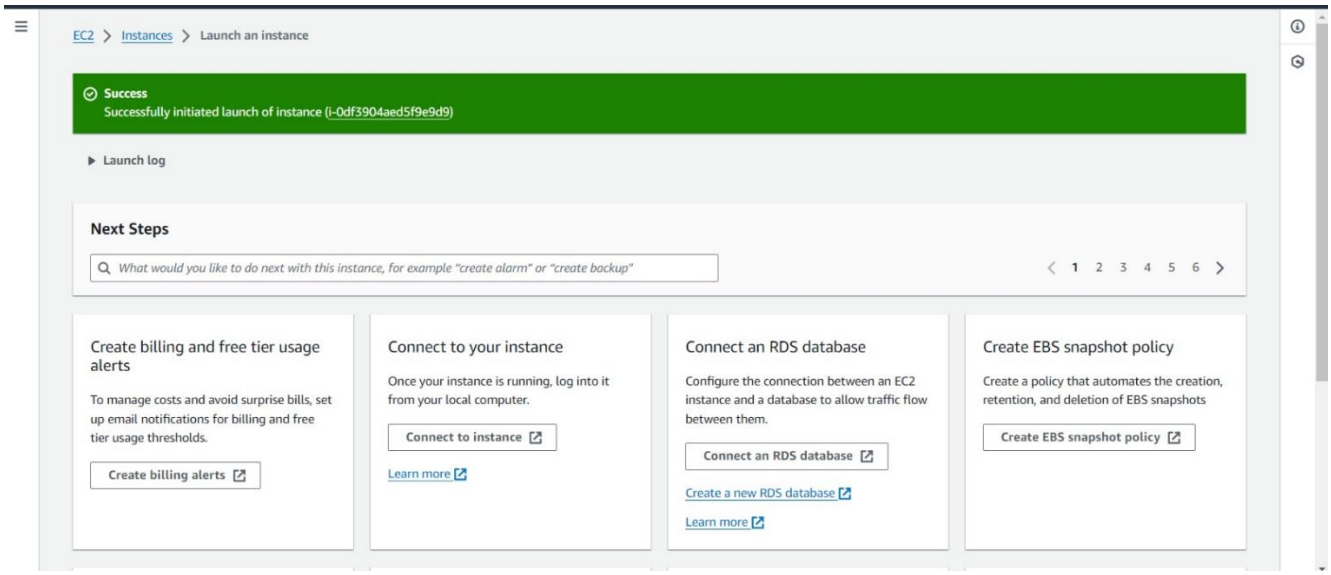


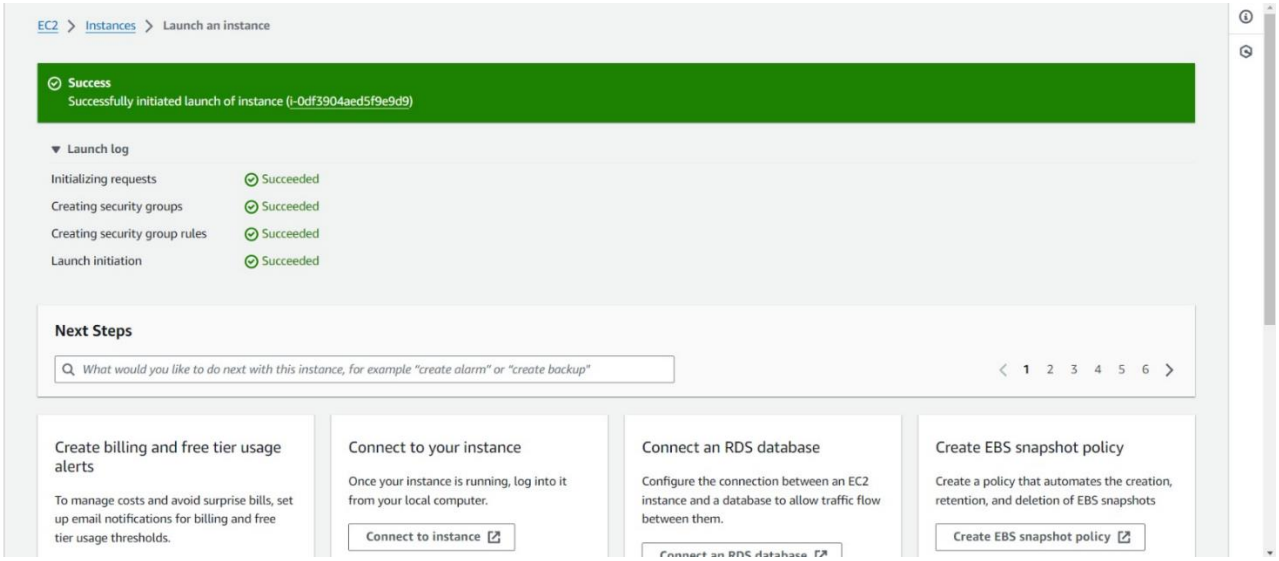
Anushka Shahane D15A 55



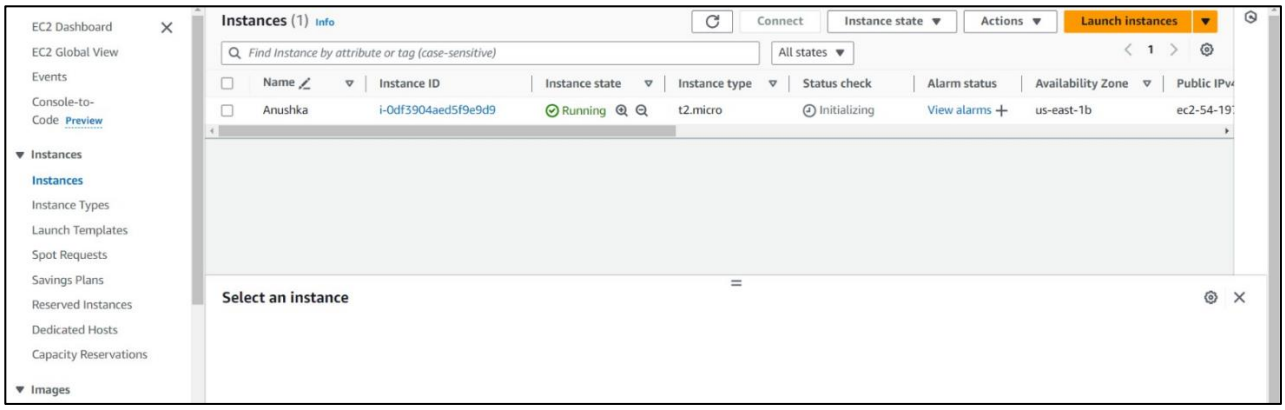


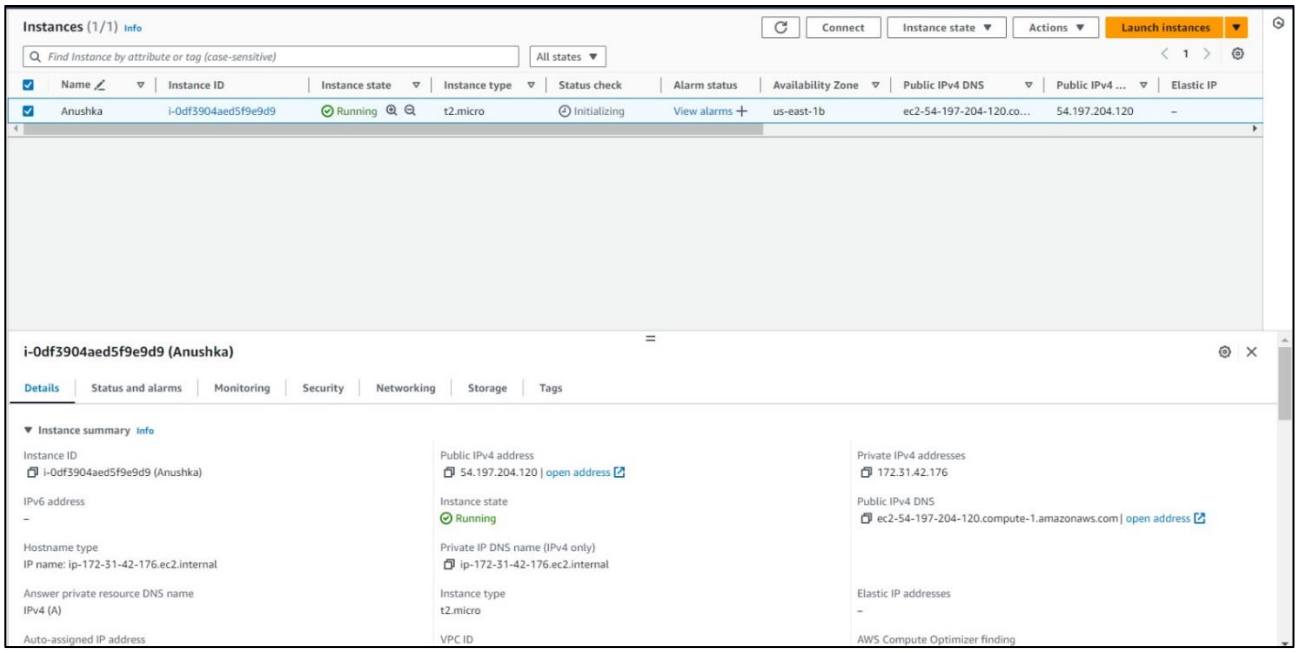
Anushka Shahane D15A 55





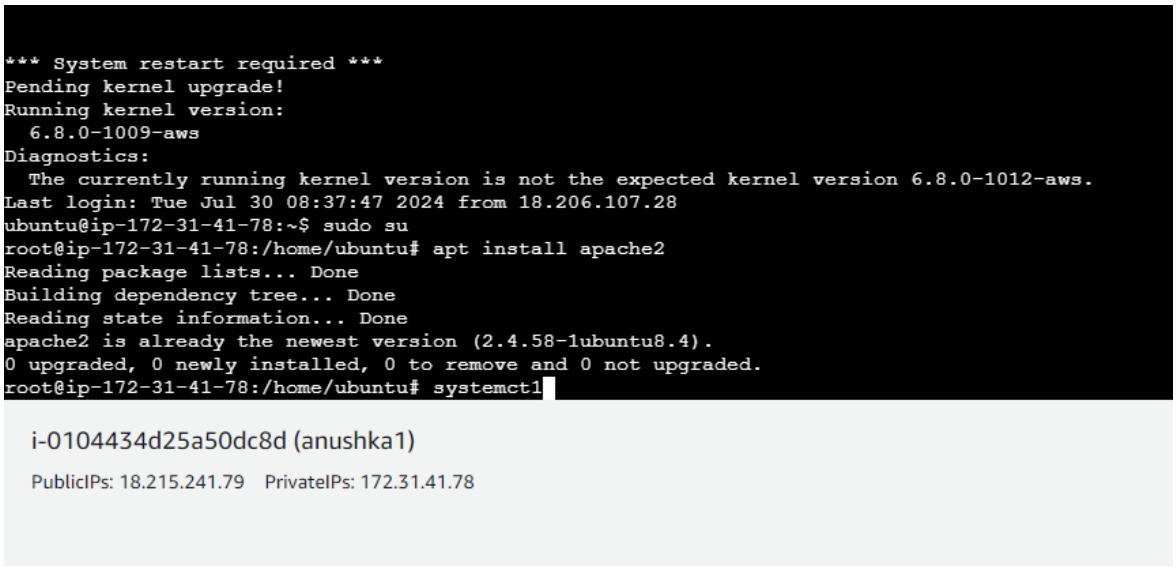
Anushka Shahane D15A 55

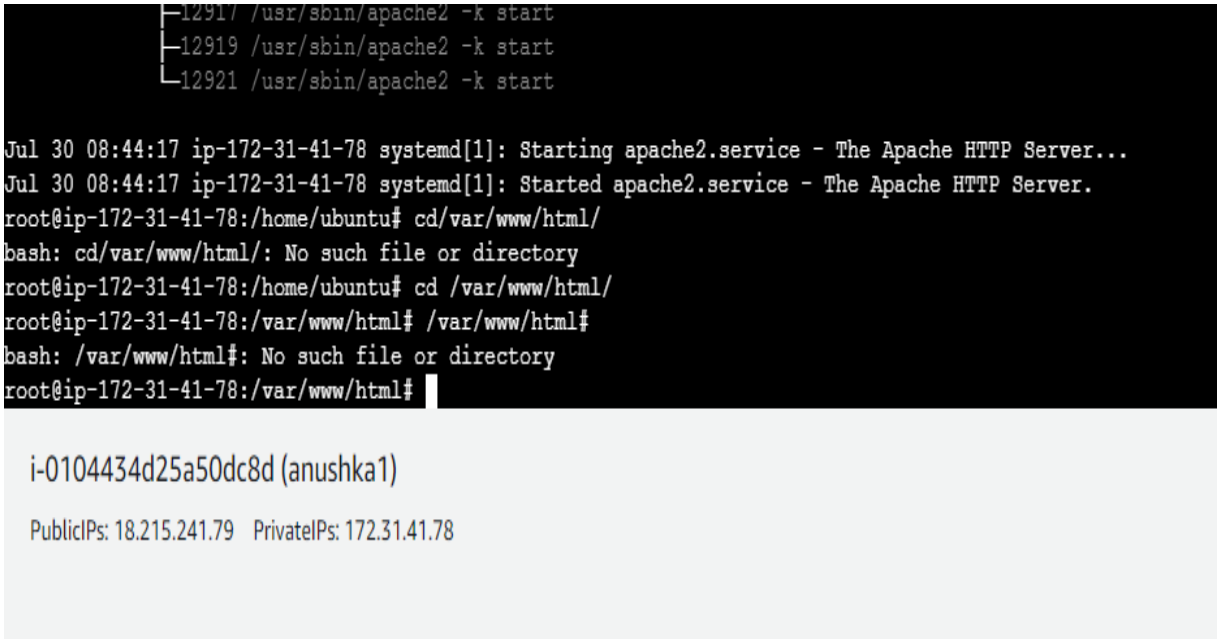





```
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-172-31-42-176:~$ ls
ubuntu@ip-172-31-42-176:~$ echo "hello"
hello
ubuntu@ip-172-31-42-176:~$ cat > myfile.txt
This is Advance devops lab
^C
ubuntu@ip-172-31-42-176:~$ cat myfile
cat: myfile: No such file or directory
ubuntu@ip-172-31-42-176:~$ cat myfile.txt
This is Advance devops lab
ubuntu@ip-172-31-42-176:~$
```

Hosting a static website using EC2 instance:



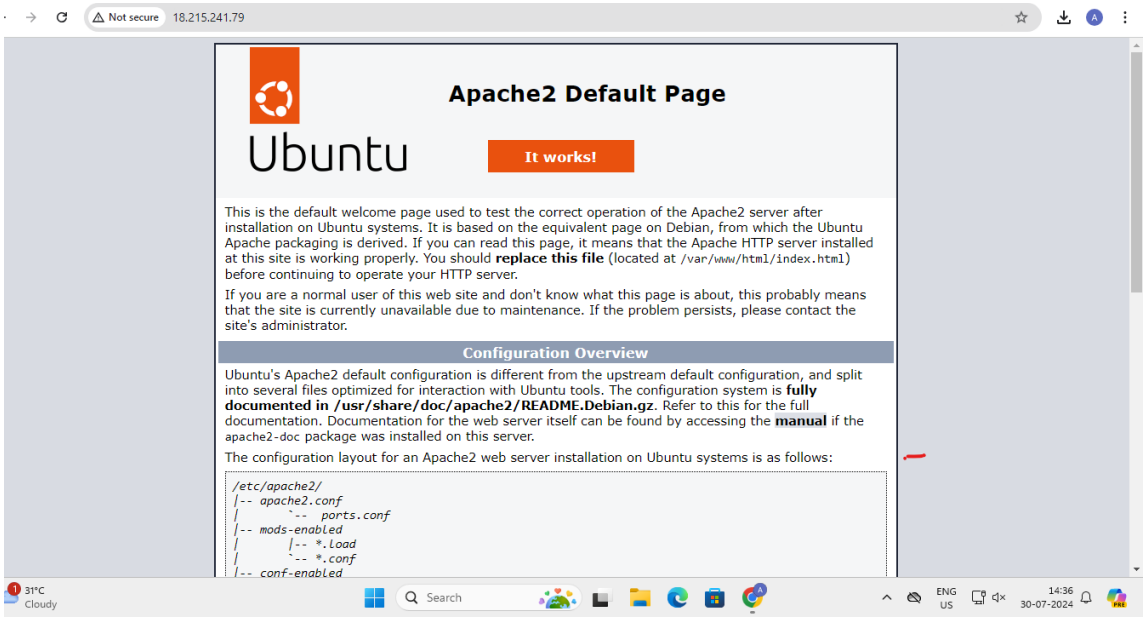


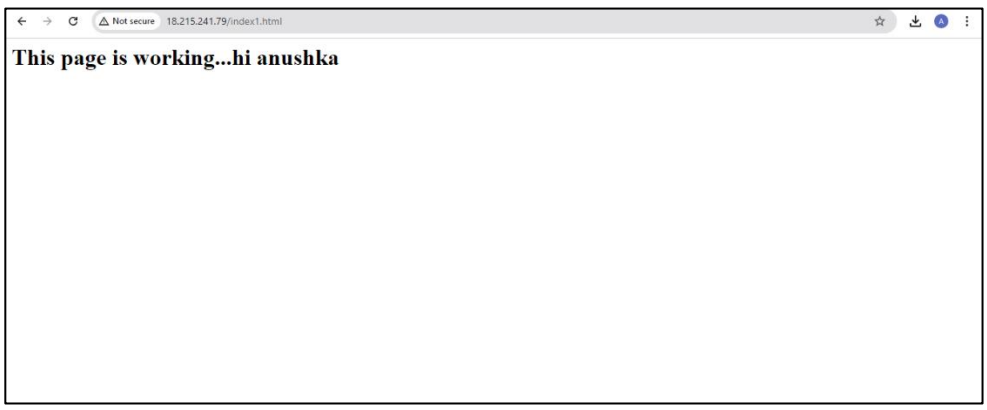
Anushka Shahane D15A 55

```
  command 'systemctl' from deb systemctl (1.4.4181-1.1)
Try: apt install <deb name>
root@ip-172-31-41-78:/home/ubuntu# systemctl status apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-07-30 08:44:17 UTC; 12min ago
       Docs: https://httpd.apache.org/docs/2.4/
   Main PID: 12917 (apache2)
      Tasks: 55 (limit: 1130)
     Memory: 5.3M (peak: 5.4M)
        CPU: 74ms
     CGroup: /system.slice/apache2.service
             ├─12917 /usr/sbin/apache2 -k start
             ├─12919 /usr/sbin/apache2 -k start
             └─12921 /usr/sbin/apache2 -k start

Jul 30 08:44:17 ip-172-31-41-78 systemd[1]: Starting apache2.service - The Apache HTTP Server...
Jul 30 08:44:17 ip-172-31-41-78 systemd[1]: Started apache2.service - The Apache HTTP Server.
root@ip-172-31-41-78:/home/ubuntu#
```

i-0104434d25a50dc8d (anushka1)

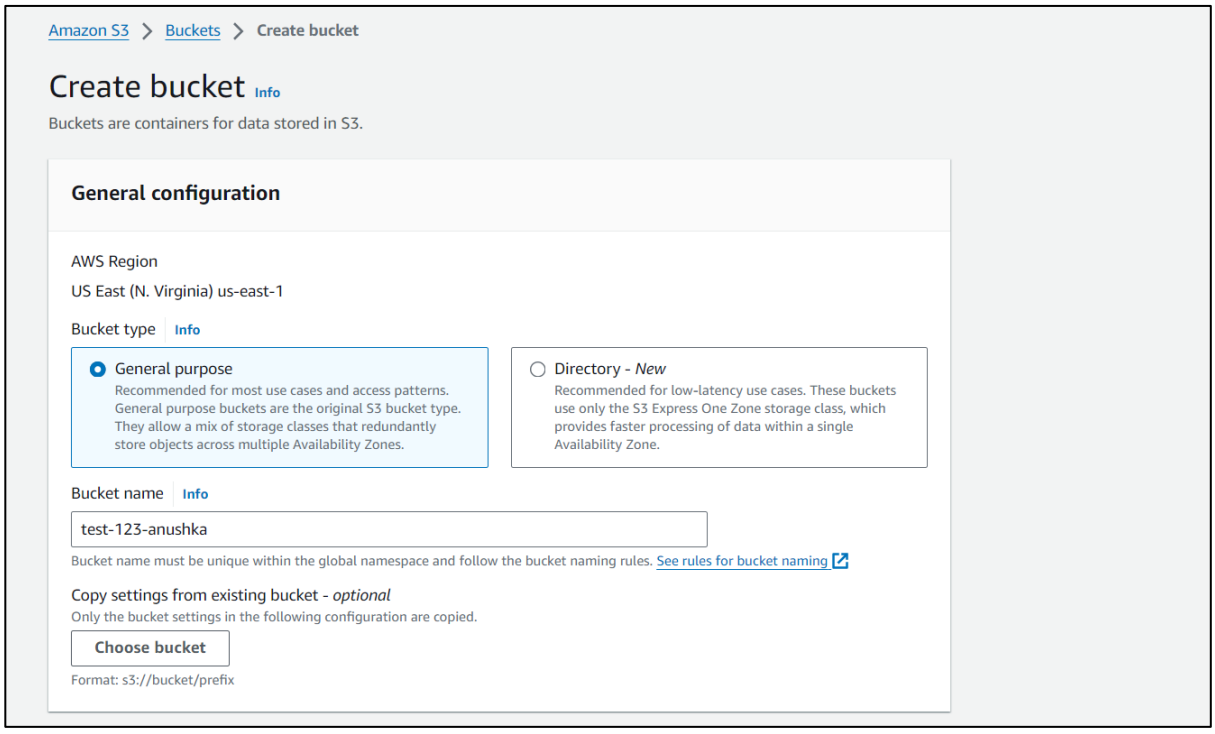
PublicIPs: 18.215.241.79 PrivateIPs: 172.31.41.78

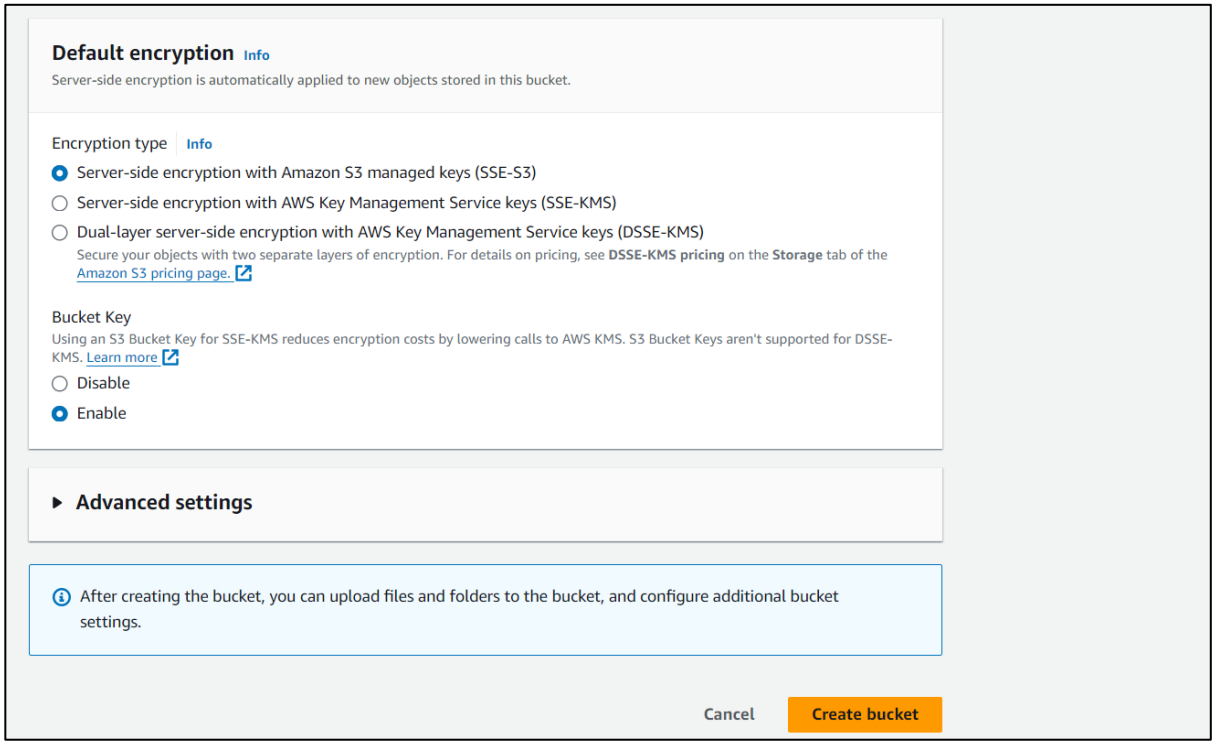




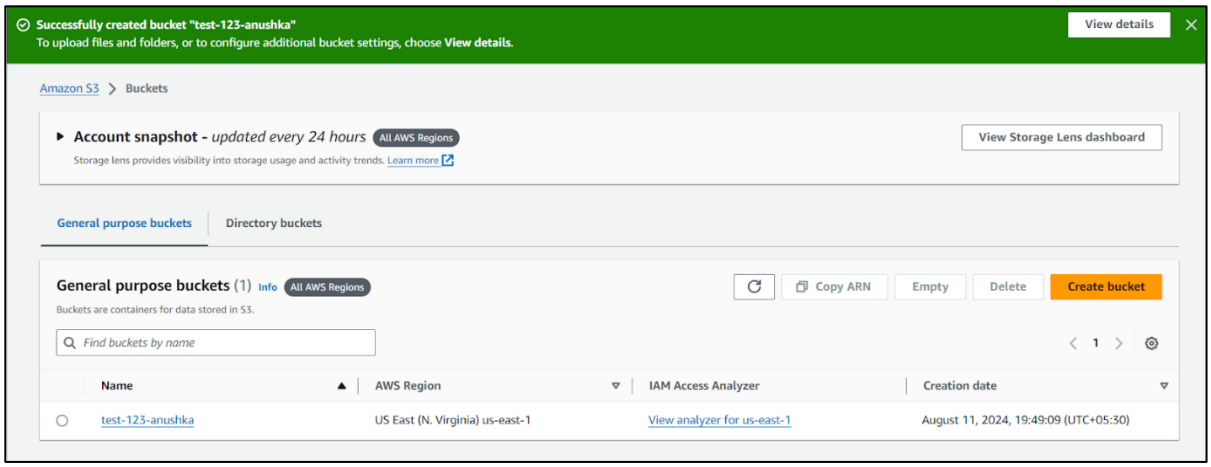
Anushka Shahane D15A 55

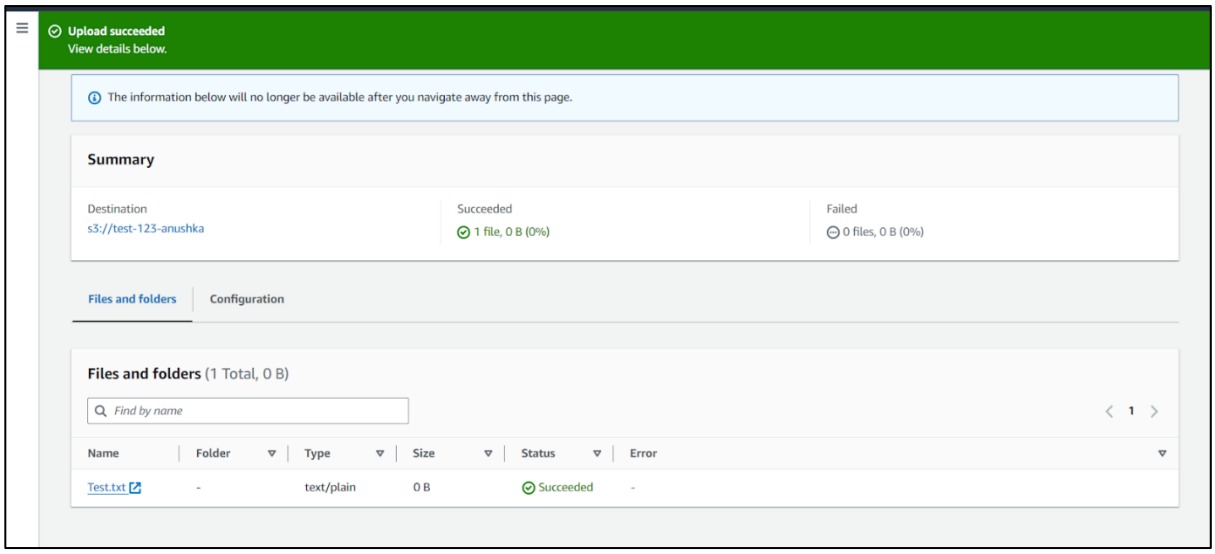
Hosting using S3 bucket :

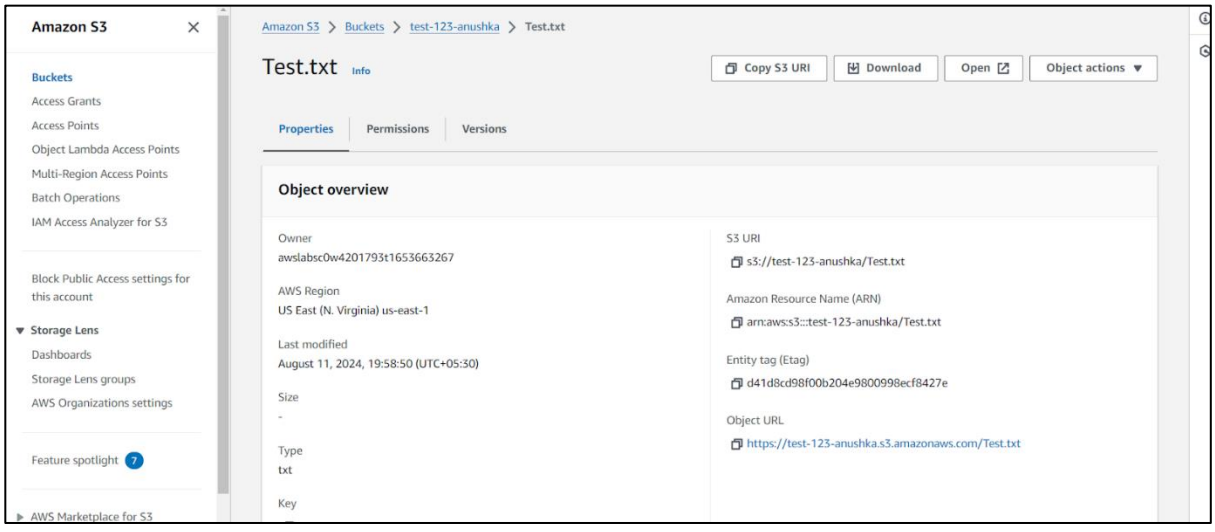


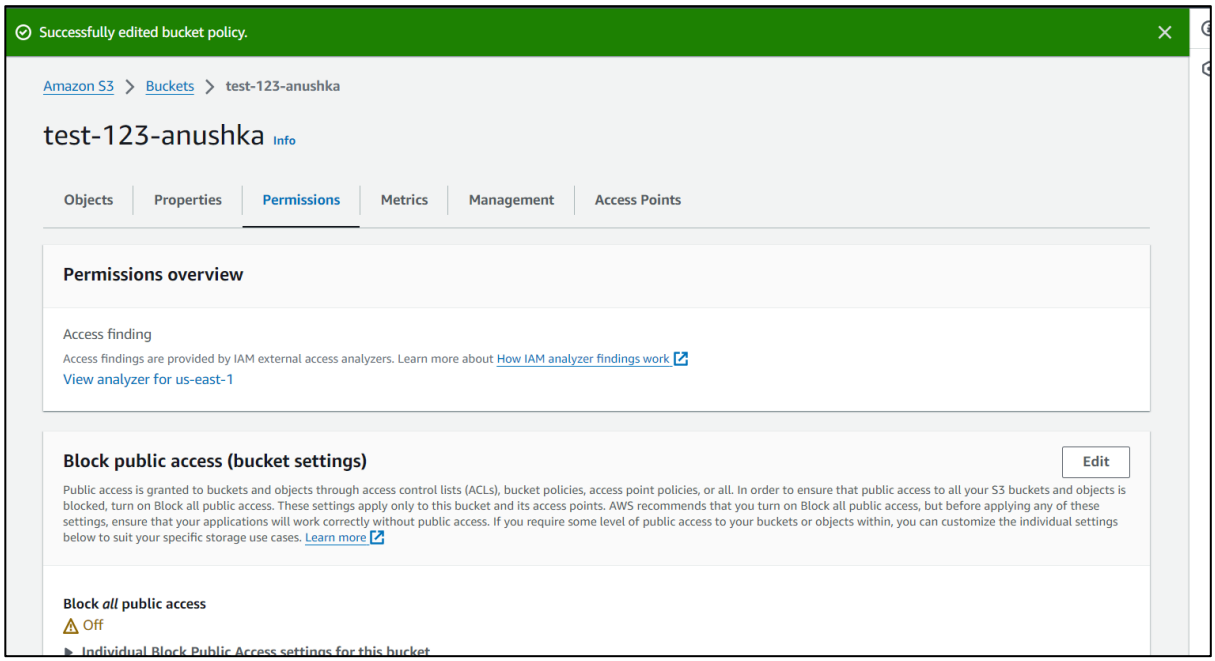


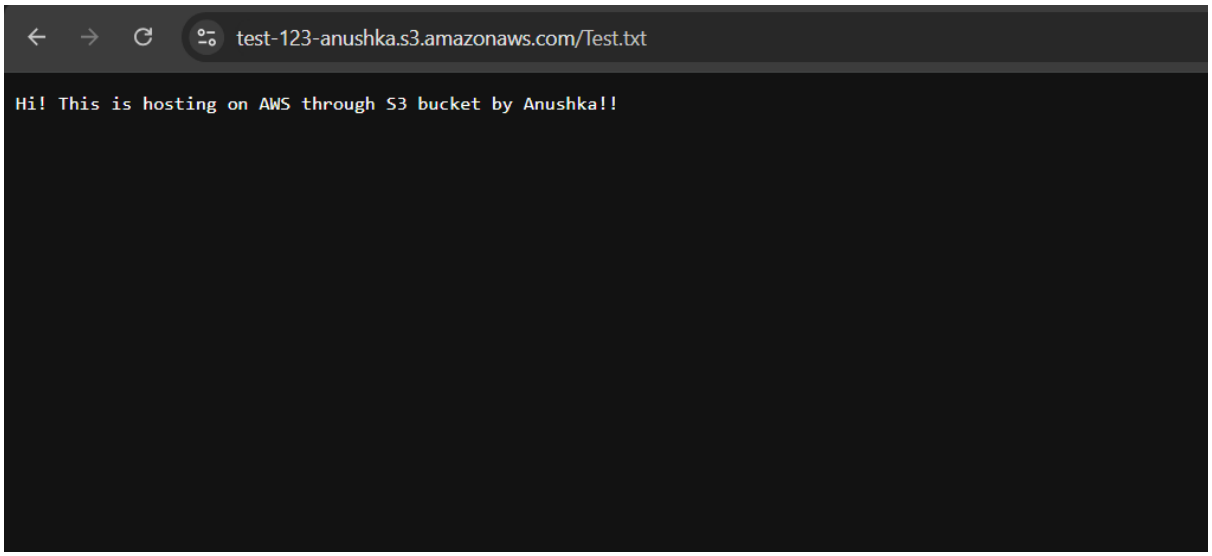
Anushka Shahane D15A 55



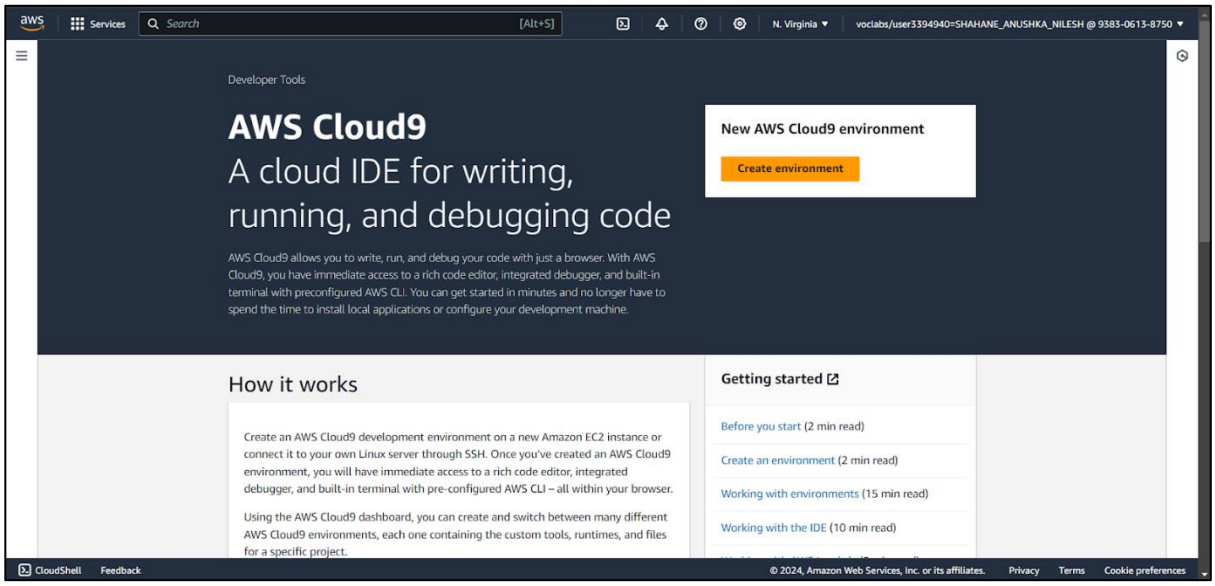


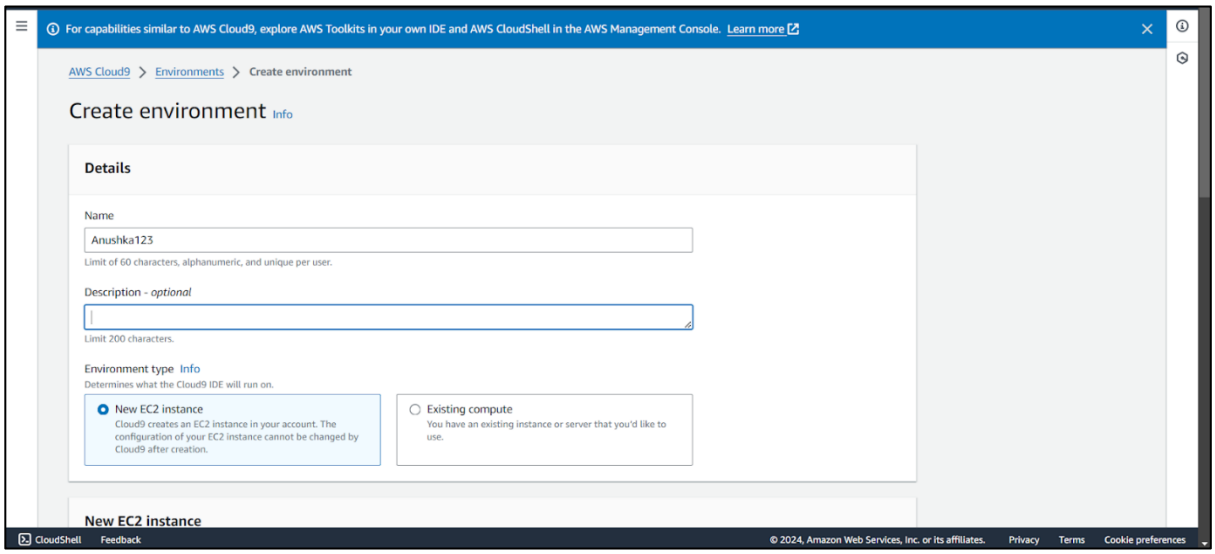


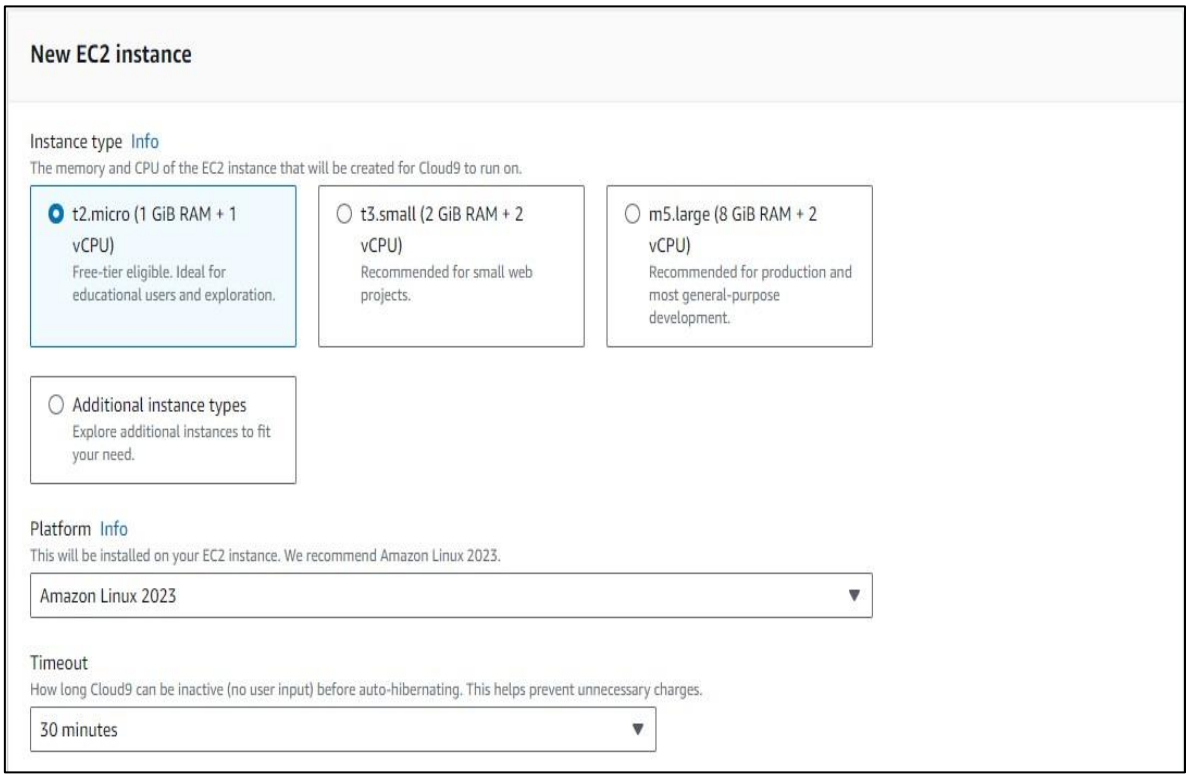

Successfully edited bucket policy.
Amazon S3 > Buckets > test-123-anushka
test-123-anushka Info
Objects
Properties
Permissions
Metrics
Management
Access Points
Permissions overview
Access finding
Access findings are provided by IAM external access analyzers. Learn more about How IAM analyzer findings work
View analyzer for us-east-1
Block public access (bucket settings)
Edit
Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. Learn more
Block all public access
Off
Individual Block Public Access settings for this bucket

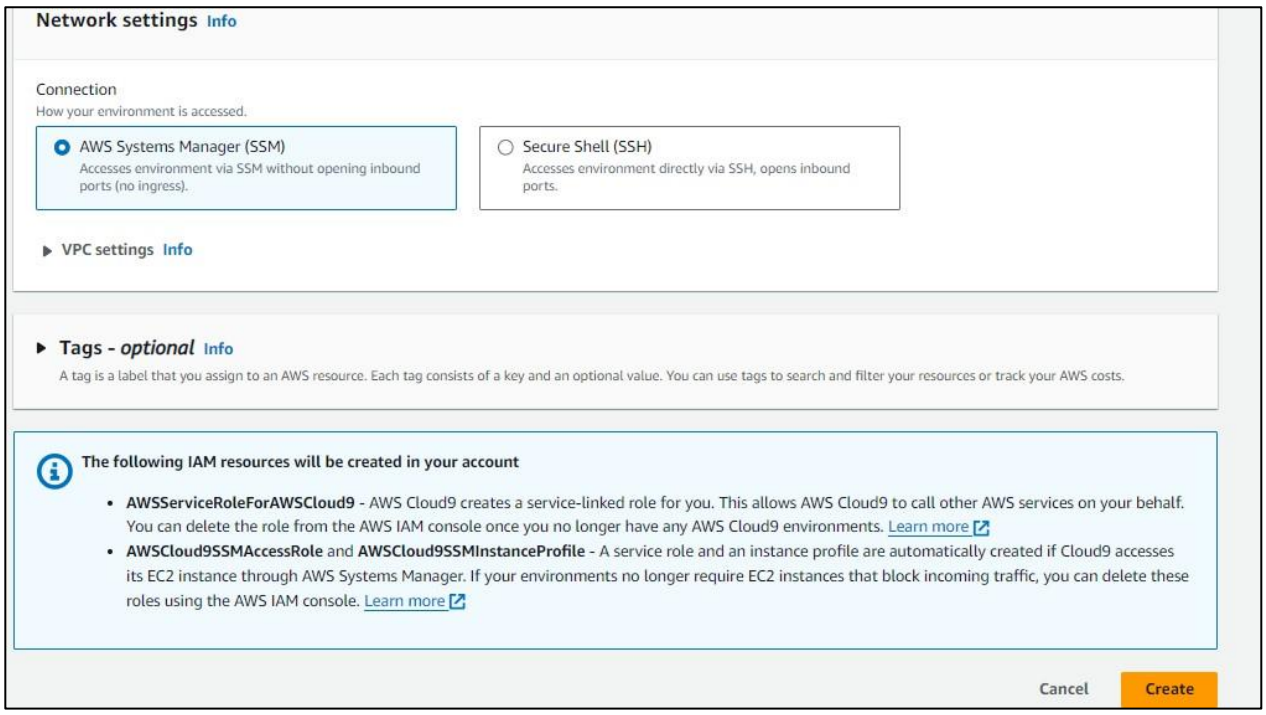

test-123-anushka.s3.amazonaws.com/Test.txt
Hi! This is hosting on AWS through S3 bucket by Anushka!!

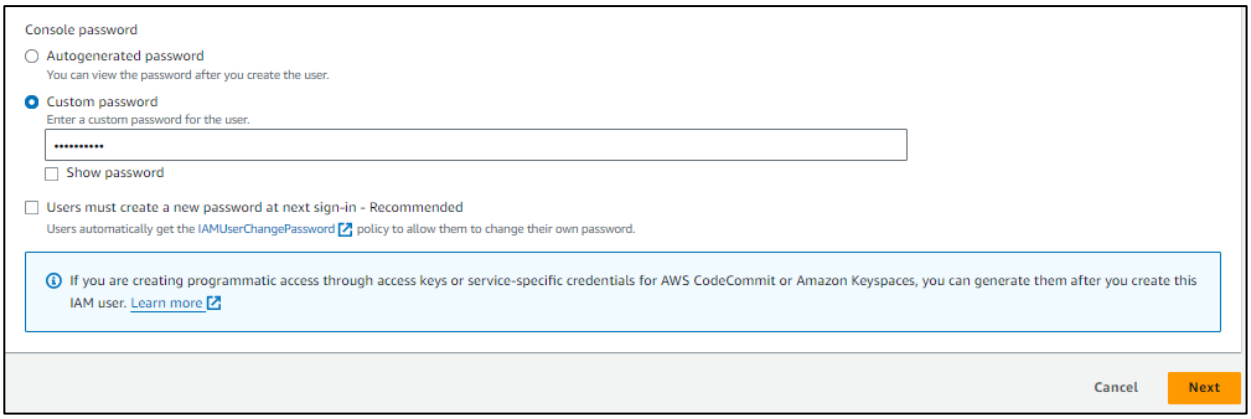
Hosting using Cloud 9 :

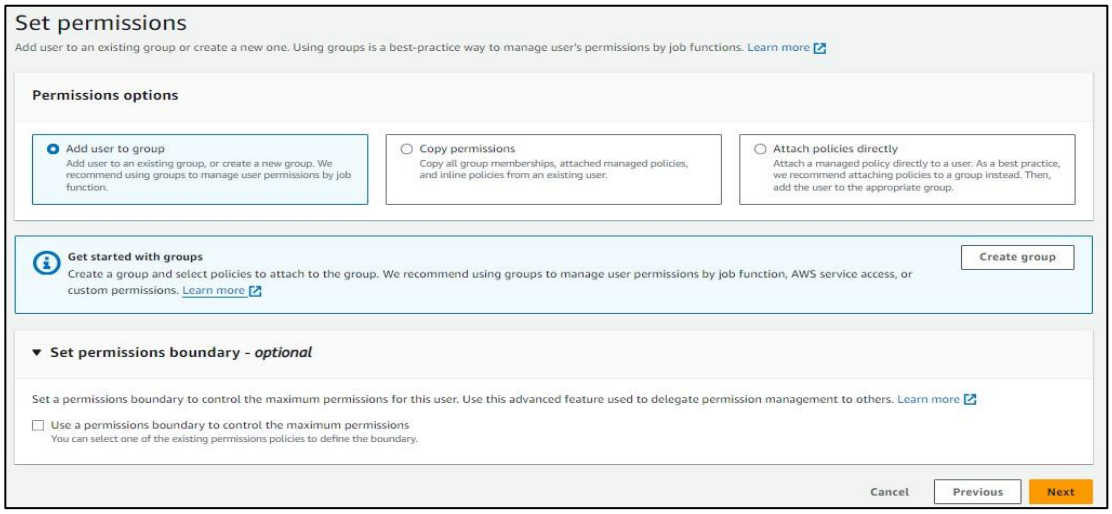




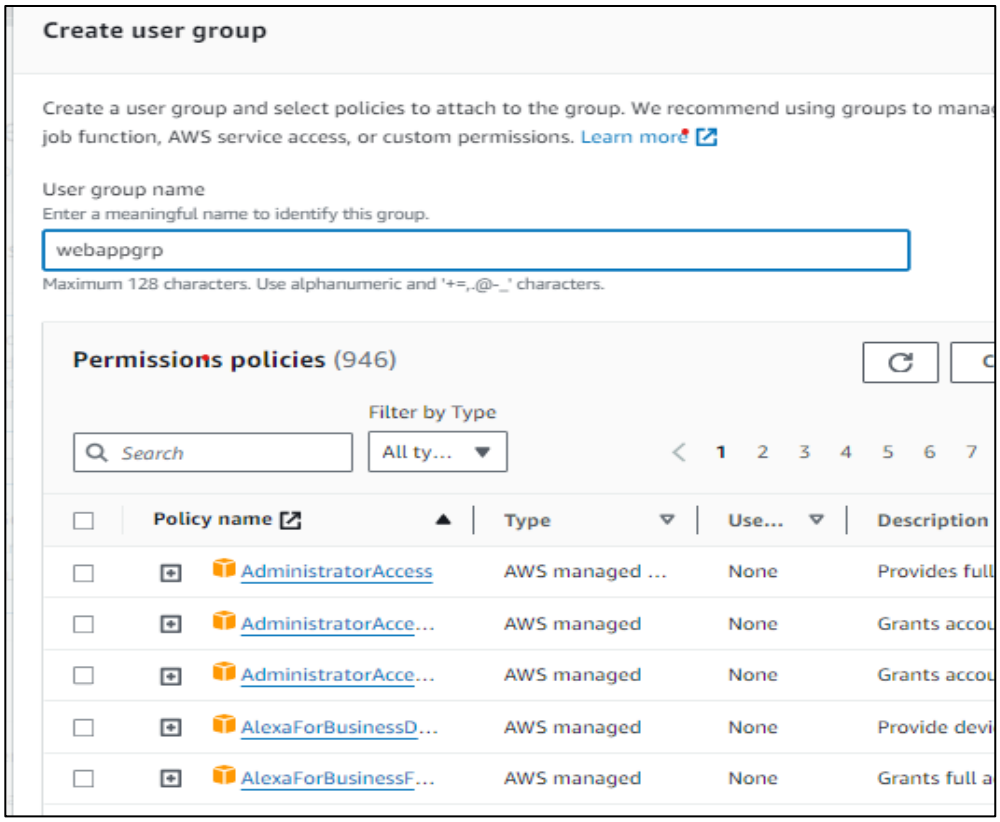

New EC2 instance
Instance type Info
The memory and CPU of the EC2 instance that will be created for Cloud9 to run on.
t2.micro (1 GiB RAM + 1 vCPU)
Free-tier eligible. Ideal for educational users and exploration.
t3.small (2 GiB RAM + 2 vCPU)
Recommended for small web projects.
m5.large (8 GiB RAM + 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

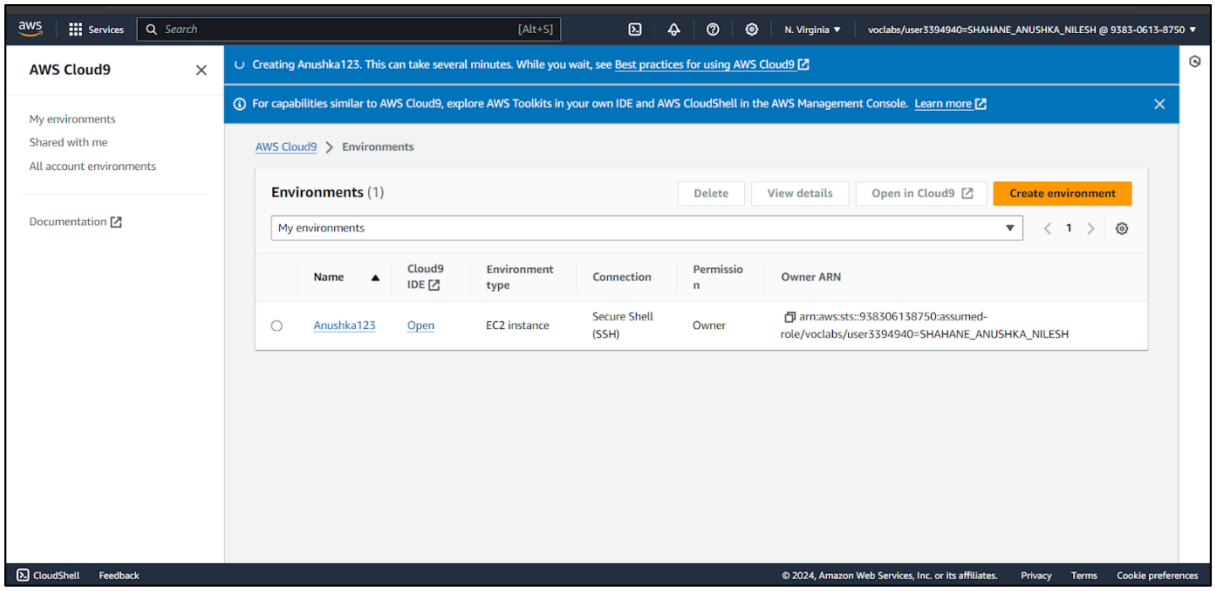

Network settings Info
Connection
How your environment is accessed.
AWS Systems Manager (SSM)
Accesses environment via SSM without opening inbound ports (no ingress).
Secure Shell (SSH)
Accesses environment directly via SSH, opens inbound ports.
VPC settings Info
Tags - optional Info
A tag is a label 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


Console password
Autogenerated password
You can view the password after you create the user.
Custom password
Enter a custom password for the user.
Show password
Users must create a new password at next sign-in - Recommended
Users automatically get the IAMUserChangePassword policy to allow them to change their own password.
If you are creating programmatic access through access keys or service-specific credentials for AWS CodeCommit or Amazon Keyspaces, you can generate them after you create this IAM user. Learn more
Cancel
Next

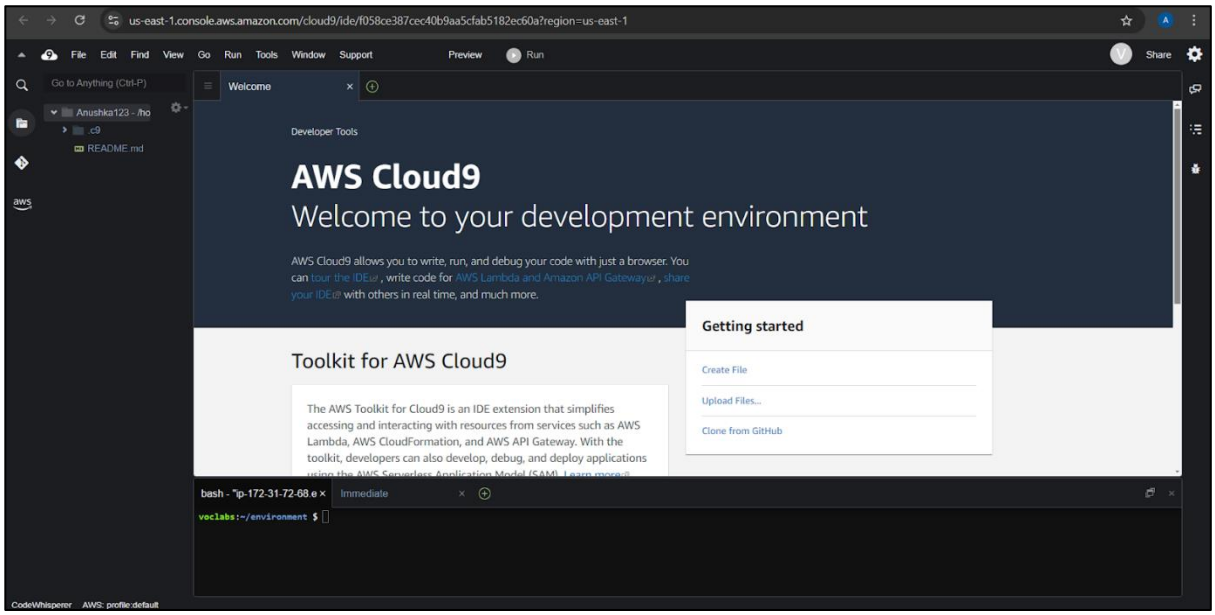

Set permissions
Add user to an existing group or create a new one. Using groups is a best-practice way to manage user's permissions by job functions. Learn more
Permissions options
Add user to group
Add user to an existing group, or create a new group. We recommend using groups to manage user permissions 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
Set a permissions boundary to control the maximum permissions for this user. Use this advanced feature used to delegate permission management to others. Learn more
Use a permissions boundary to control the maximum permissions
You can select one of the existing permissions policies to define the boundary.
Cancel
Previous
Next

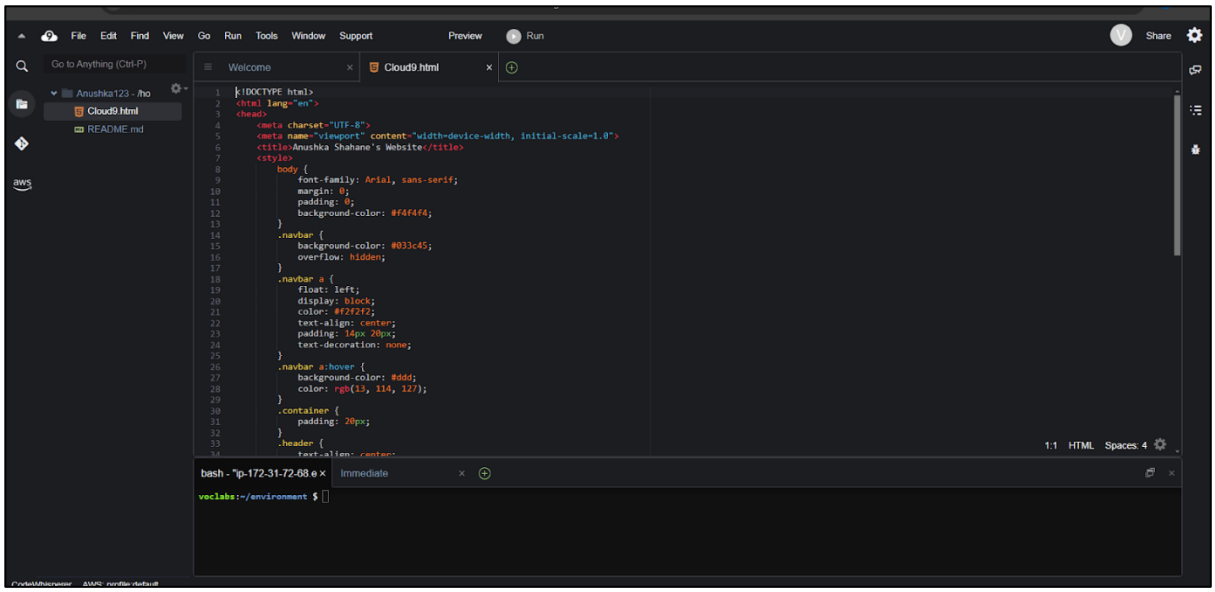
Anushka Shahane D15A 55



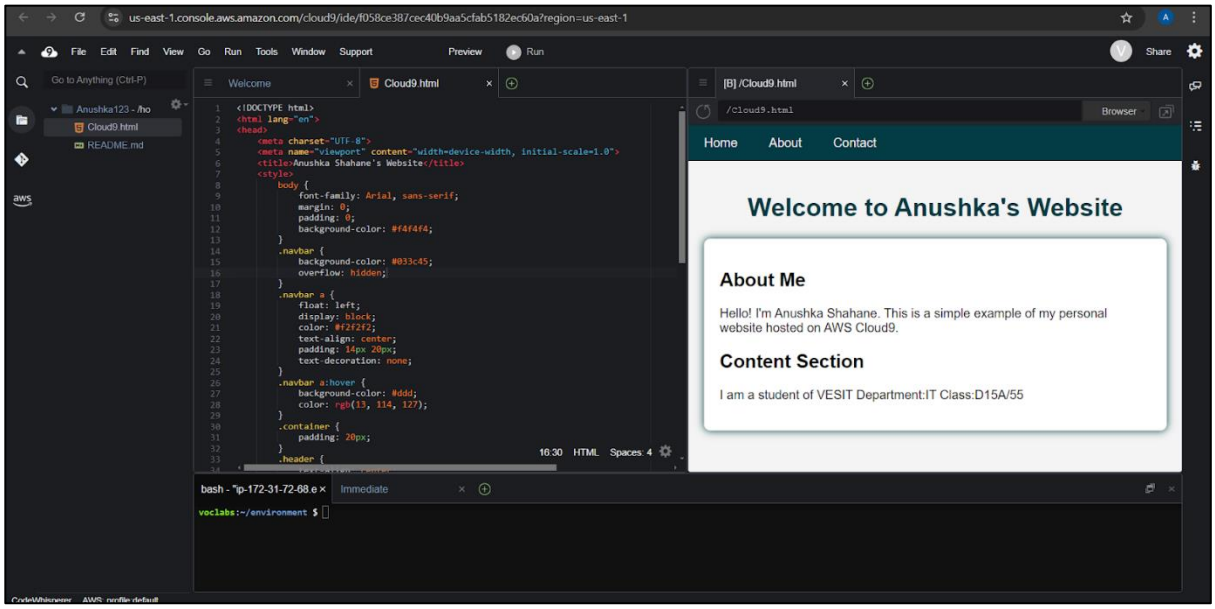


Anushka Shahane D15A 55

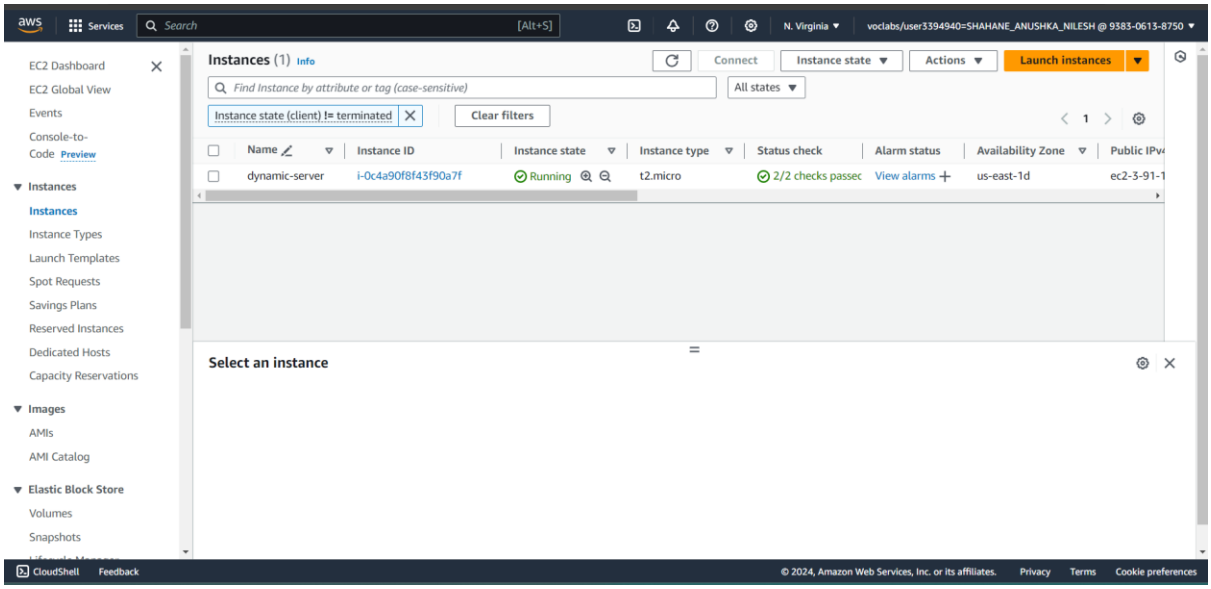




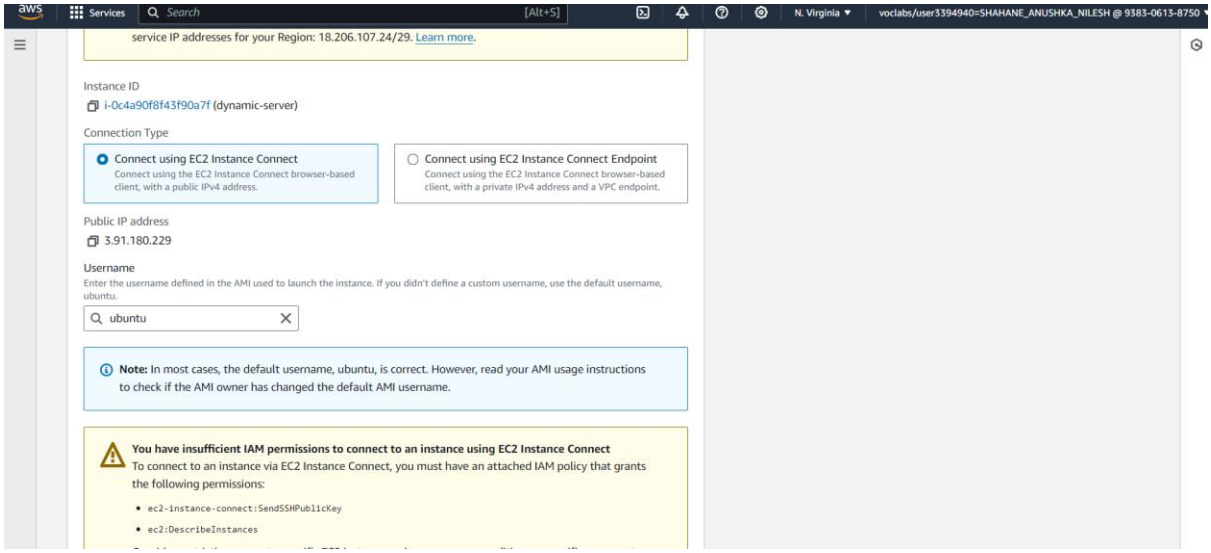
Anushka Shahane D15A 55



Dynamic Hosting using EC2 instance :



Anushka Shahane D15A 55

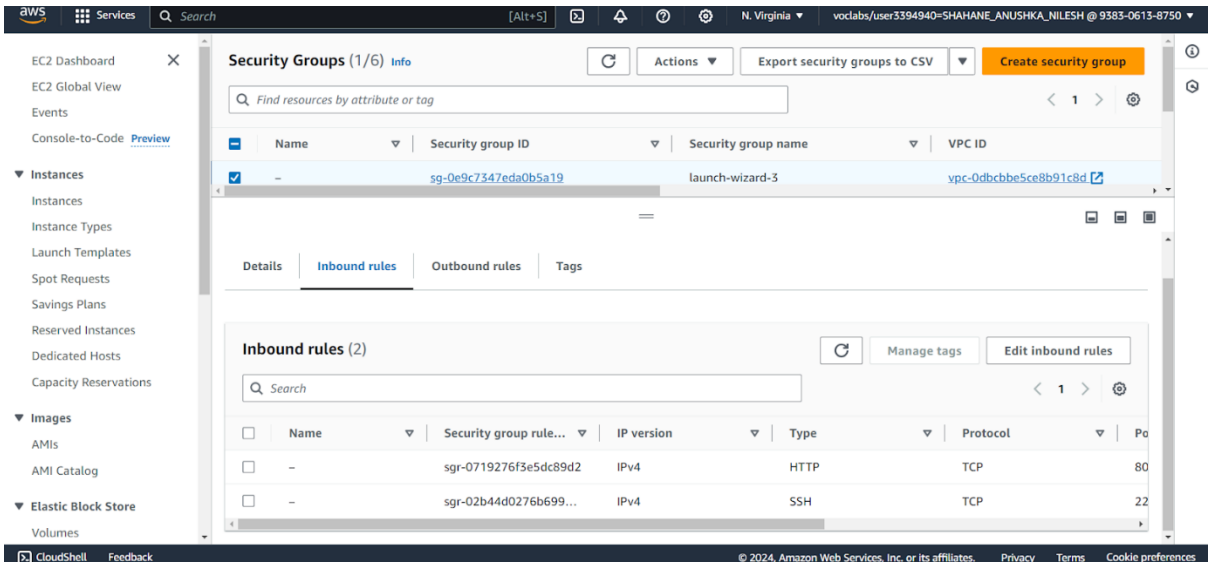


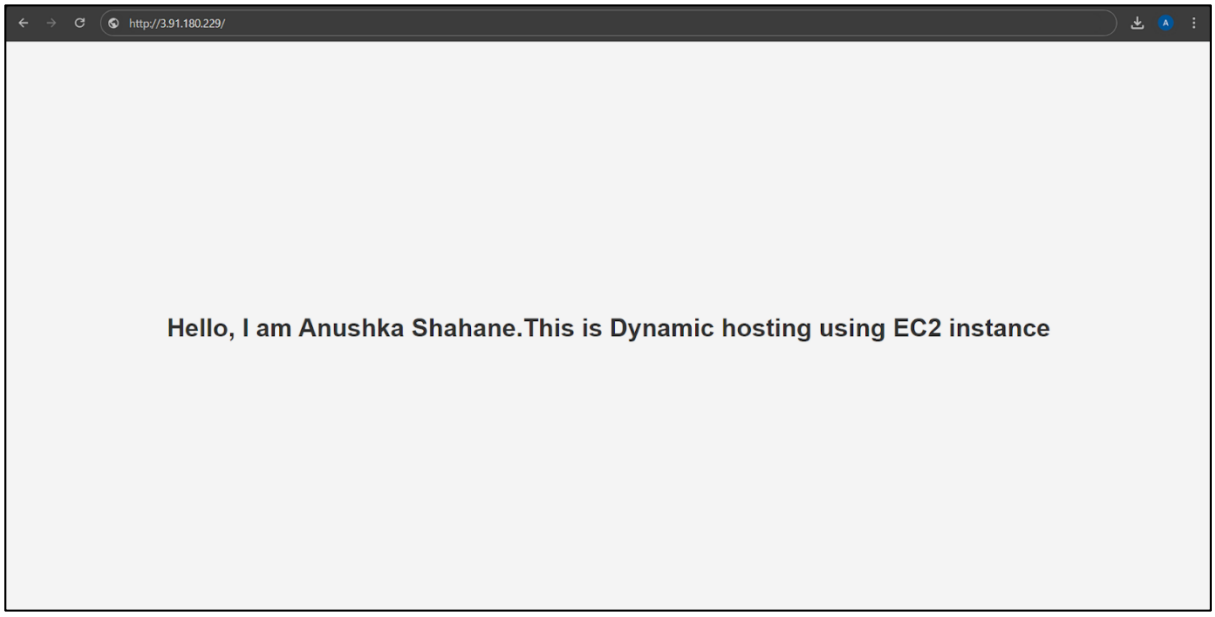

```
ubuntu@ip-172-31-83-228:~$ pwd
/home/ubuntu
ubuntu@ip-172-31-83-228:~$ mkdir Anushka
ubuntu@ip-172-31-83-228:~$ cd Anushka
ubuntu@ip-172-31-83-228:~/Anushka$ git clone https://github.com/Anushka3204/Dynamic_hosting_EC2.git
Cloning into 'Dynamic_hosting_EC2'...
remote: Enumerating objects: 5, done.
remote: Counting objects: 100% (5/5), done.
remote: Compressing objects: 100% (5/5), done.
remote: Total 5 (delta 0), reused 0 (delta 0), pack-reused 0
Receiving objects: 100% (5/5), 8.39 KiB | 1.68 MiB/s, done.
ubuntu@ip-172-31-83-228:~/Anushka$ ls
Dynamic_hosting_EC2
ubuntu@ip-172-31-83-228:~/Anushka$ cd^C
ubuntu@ip-172-31-83-228:~/Anushka$ cd Dynamic_hosting_EC2
ubuntu@ip-172-31-83-228:~/Anushka/Dynamic_hosting_EC2$ ls
index.js  package-lock.json  package.json
```

```
No VM guests are running outdated hypervisor (qemu) binaries on this host.
ubuntu@ip-172-31-83-228:~/Anushka/Dynamic_hosting_EC2/Dynamic_hosting_EC2$ npm i

added 64 packages, and audited 65 packages in 3s

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
ubuntu@ip-172-31-83-228:~/Anushka/Dynamic_hosting_EC2/Dynamic_hosting_EC2$
```

```
Dynamic_hosting_EC2  Dynamic_hosting_ec2  Hosting_dynamic_ec2  package-lock.json
ubuntu@ip-172-31-83-228:~/Anushka$ cd  Hosting_dynamic_ec2
ubuntu@ip-172-31-83-228:~/Anushka/Hosting_dynamic_ec2$ ls
index.js  package-lock.json  package.json
ubuntu@ip-172-31-83-228:~/Anushka/Hosting_dynamic_ec2$ npm i

added 64 packages, and audited 65 packages in 1s

12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities
ubuntu@ip-172-31-83-228:~/Anushka/Hosting_dynamic_ec2$ npm start

> server@1.0.0 start
> node index.js

Server is running on port 3000
```

Anushka Shahane D15A 55





Anushka Shahane D15A 55

Anushka Shahane D15A 55