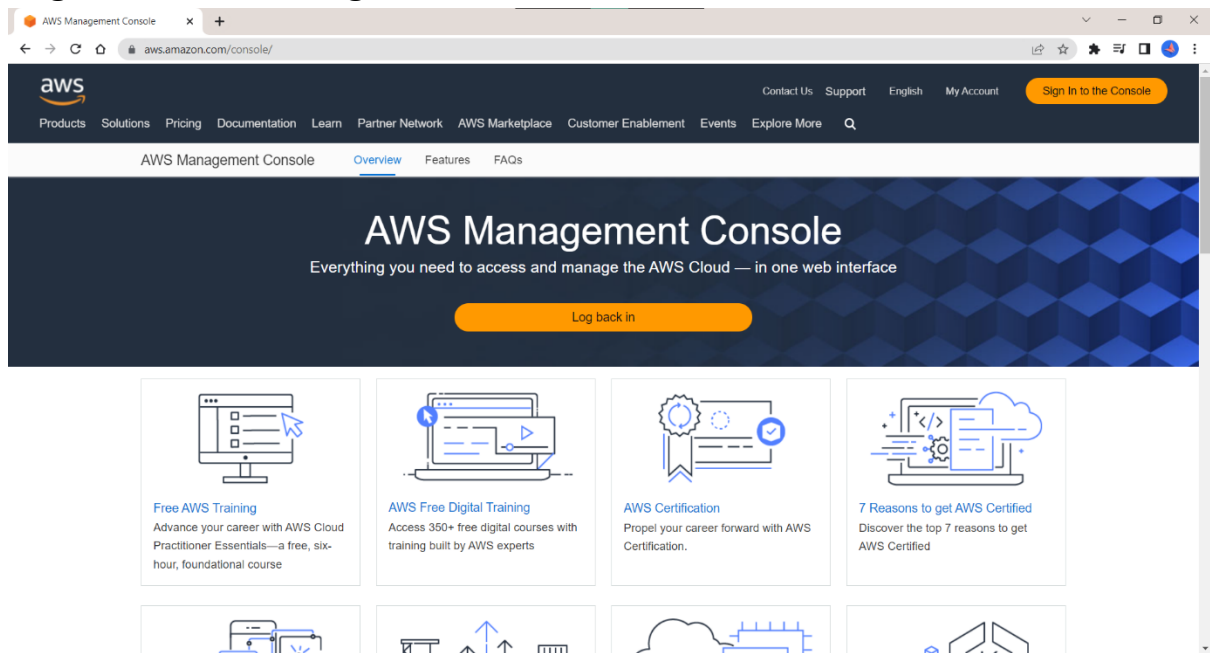
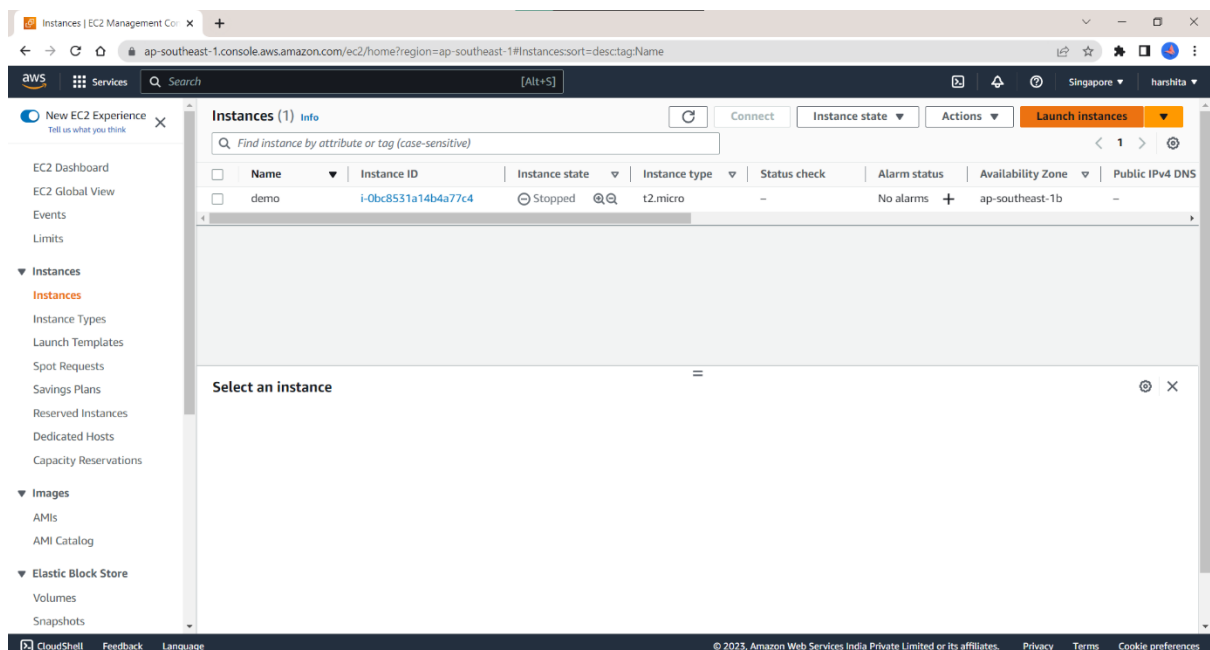


# OUTPUTS:

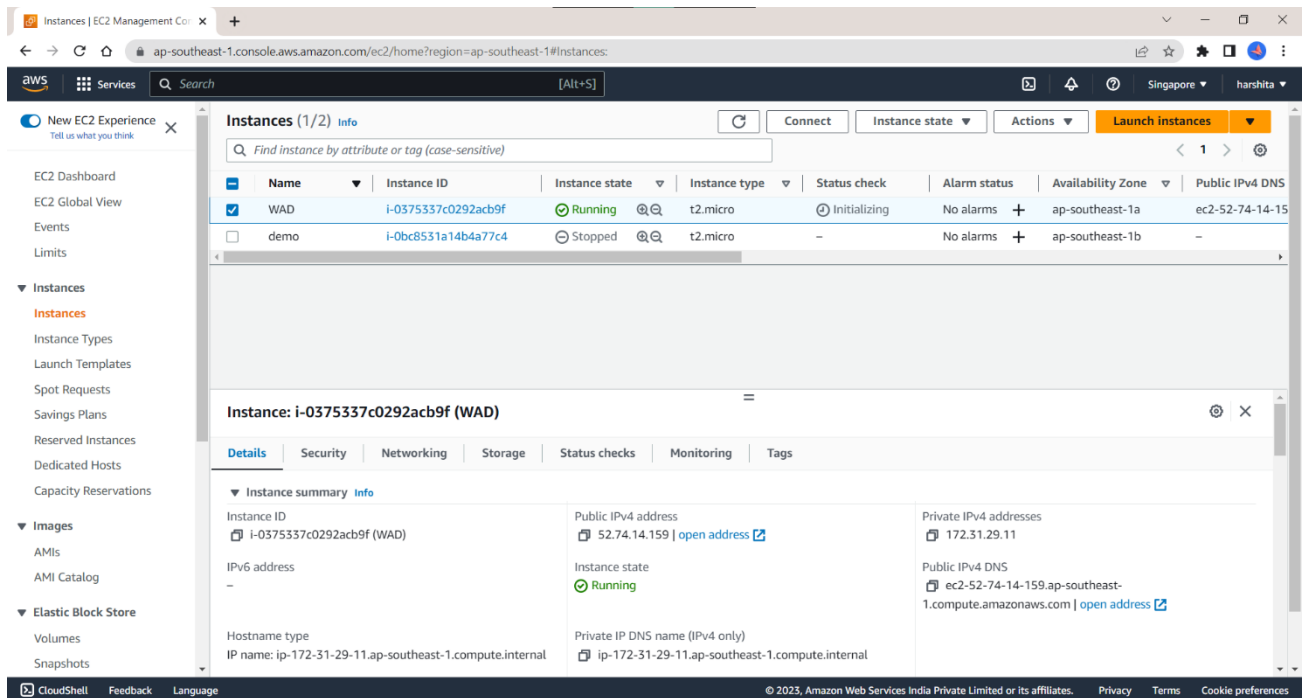
## Login to AWS Management Console:



## Create an EC2 Instance:

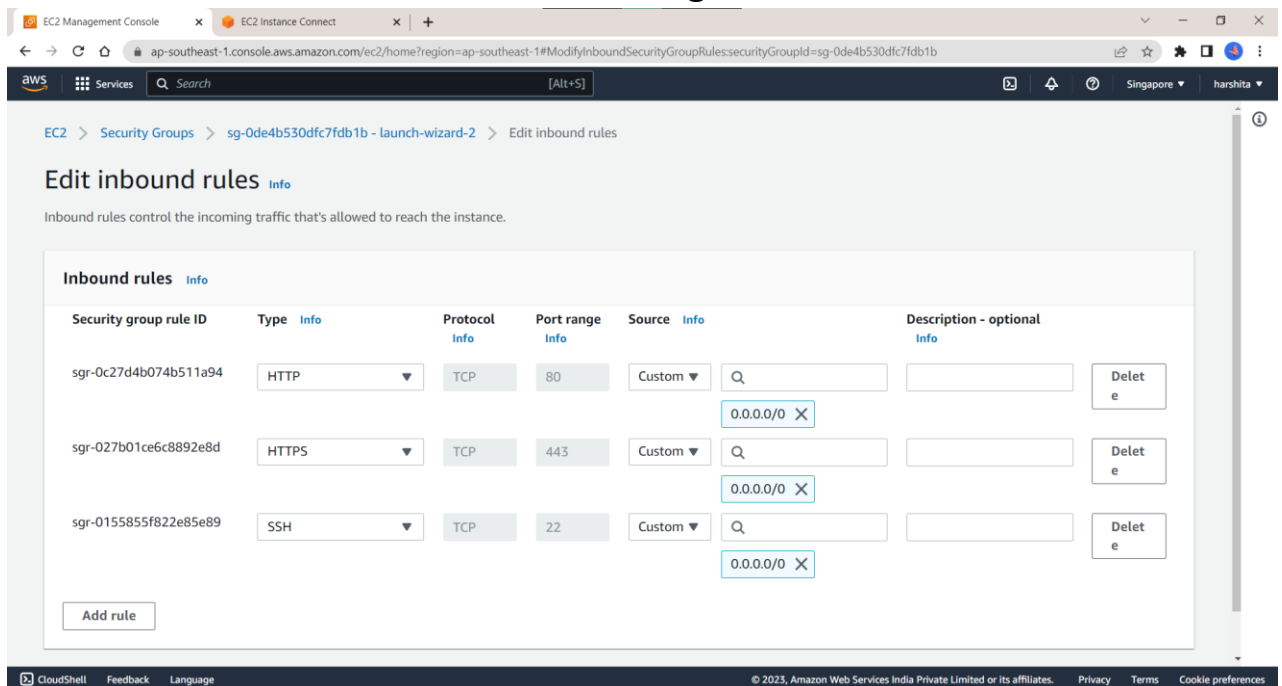


## We created an instance name WAD:



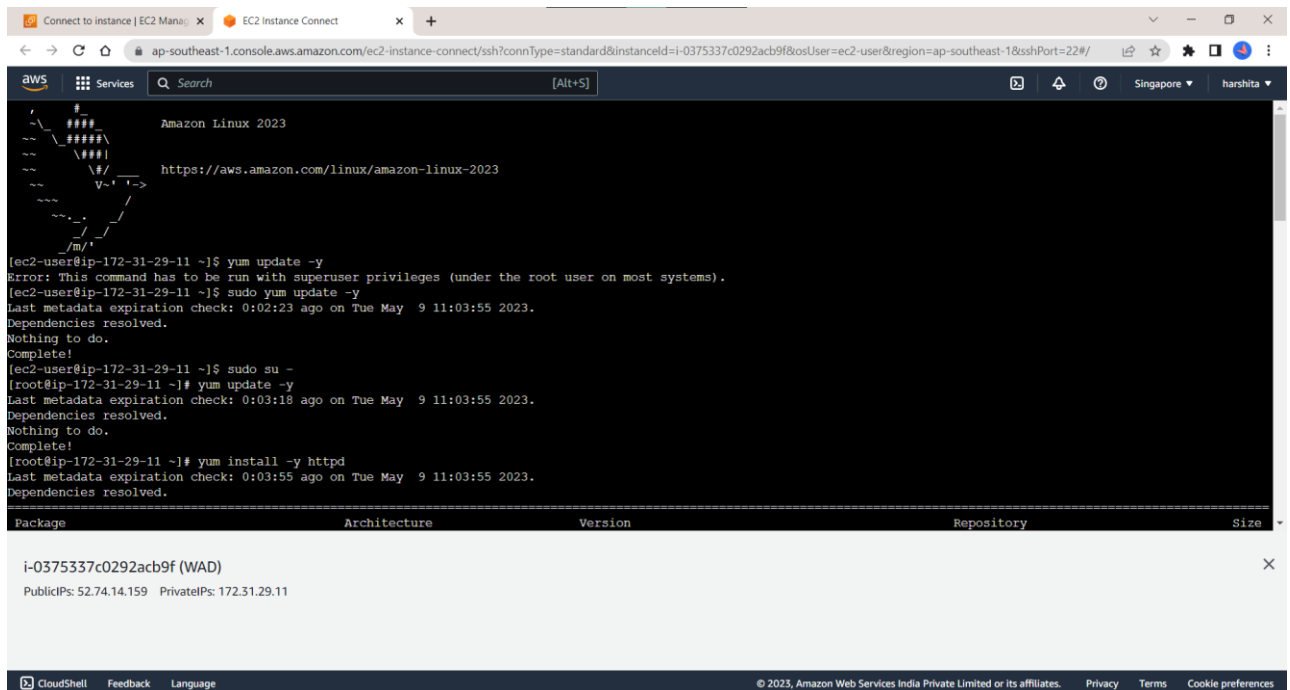
The screenshot displays the AWS Management Console for the EC2 service. The left sidebar shows the navigation menu with options like EC2 Dashboard, EC2 Global View, Events, Limits, Instances, Instance Types, Launch Templates, Spot Requests, Savings Plans, Reserved Instances, Dedicated Hosts, Capacity Reservations, Images, AMIs, AMI Catalog, Elastic Block Store, Volumes, and Snapshots. The main content area shows the 'Instances (1/2) Info' page. A table lists the instances, with 'WAD' (ID: i-0375337c0292acb9f) in the 'Running' state. Below the table, the details for instance 'WAD' are shown, including its Instance ID, Public IPv4 address (52.74.14.159), Private IPv4 addresses (172.31.29.11), Instance state (Running), Public IPv4 DNS (ec2-52-74-14-159.ap-southeast-1.compute.amazonaws.com), Private IP DNS name (ip-172-31-29-11.ap-southeast-1.compute.internal), and Hostname type (ip-172-31-29-11.ap-southeast-1.compute.internal).

## Edit the inbound rules to allow incoming traffic:



The screenshot displays the AWS Management Console for the EC2 service, specifically the 'Edit inbound rules' page for the security group 'sg-0de4b530dfc7fdb1b'. The page shows three inbound rules: HTTP (port 80), HTTPS (port 443), and SSH (port 22). Each rule is configured to allow traffic from 0.0.0.0/0. The 'Add rule' button is visible at the bottom left.

Connect the instance using EC2 Connect and login as root user:



```
#
#####
      _   _
     / \ / \
    /   /   \
   /___/_____\
  /   /   \   \
 /___/_____\   \
/   /   \   \   \
/_/_/_____\   \_/_/
/m/  \_/_/

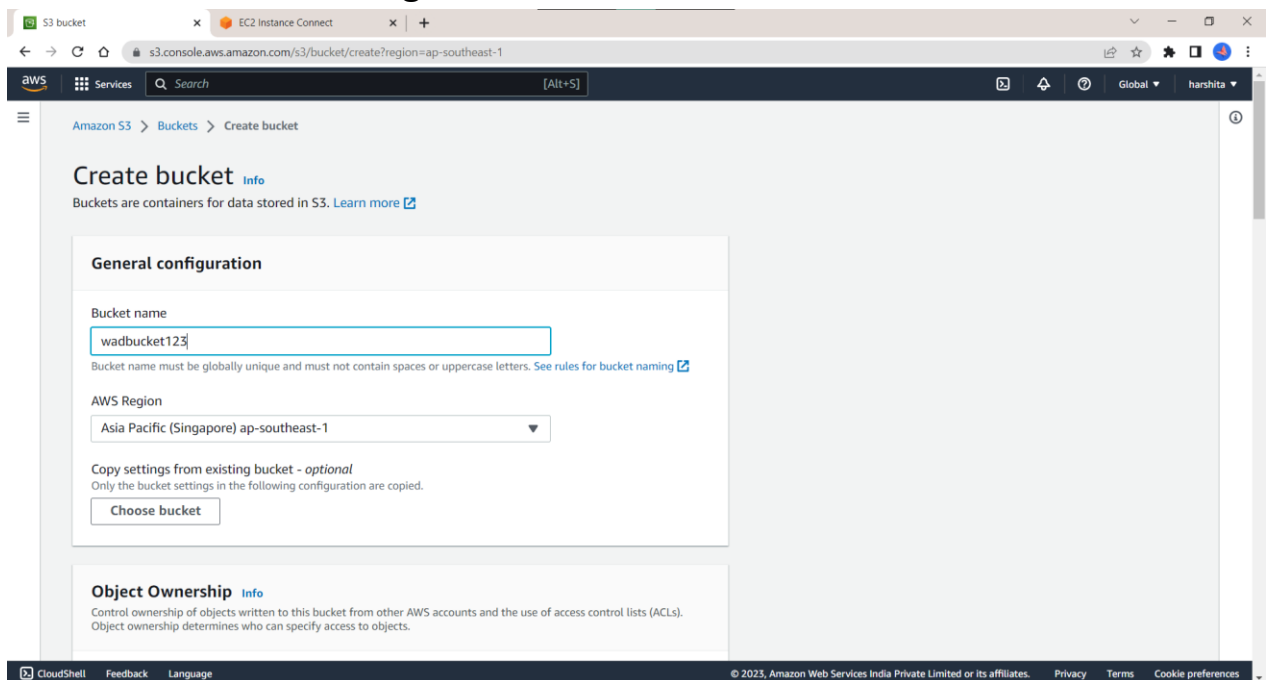
Amazon Linux 2023

https://aws.amazon.com/linux/amazon-linux-2023

[ec2-user@ip-172-31-29-11 ~]$ yum update -y
Error: This command has to be run with superuser privileges (under the root user on most systems).
[ec2-user@ip-172-31-29-11 ~]$ sudo yum update -y
Last metadata expiration check: 0:02:23 ago on Tue May  9 11:03:55 2023.
Dependencies resolved.
Nothing to do.
Complete!
[ec2-user@ip-172-31-29-11 ~]$ sudo su -
[root@ip-172-31-29-11 ~]# yum update -y
Last metadata expiration check: 0:03:18 ago on Tue May  9 11:03:55 2023.
Dependencies resolved.
Nothing to do.
Complete!
[root@ip-172-31-29-11 ~]# yum install -y httpd
Last metadata expiration check: 0:03:55 ago on Tue May  9 11:03:55 2023.
Dependencies resolved.
```

Package	Architecture	Version	Repository	Size
i-0375337c0292acb9f (WAD)				
PublicIPs: 52.74.14.159 PrivateIPs: 172.31.29.11				

Create bucket for storage of static website:



**Create bucket** [Info](#)

Buckets are containers for data stored in S3. [Learn more](#)

**General configuration**

Bucket name

Bucket name must be globally unique and must not contain spaces or uppercase letters. [See rules for bucket naming](#)

AWS Region

Asia Pacific (Singapore) ap-southeast-1

Copy settings from existing bucket - *optional*

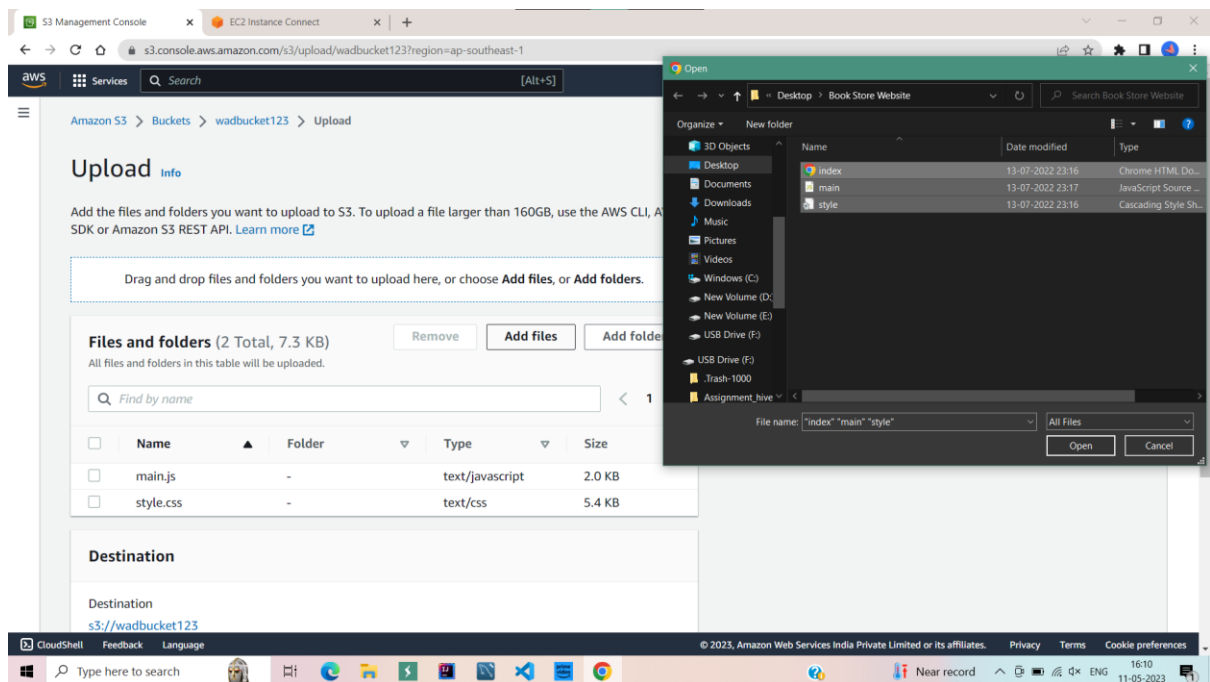
Only the bucket settings in the following configuration are copied.

[Choose bucket](#)

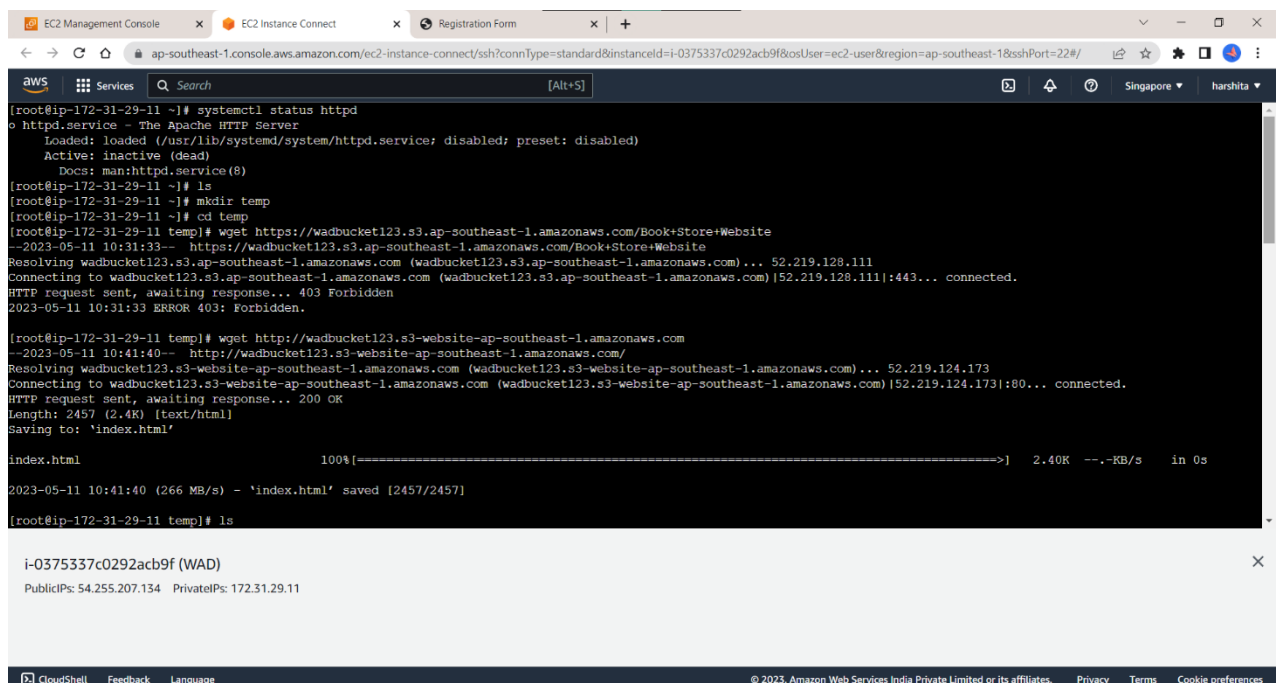
**Object Ownership** [Info](#)

Control ownership of objects written to this bucket from other AWS accounts and the use of access control lists (ACLs). Object ownership determines who can specify access to objects.

Add all the files into the bucket created:



Install the Apache server and make a directory named as “temp” and get all static files of website from S3 bucket created:



The screenshot shows a terminal window in the AWS CloudShell environment. The user is at the root of an EC2 instance with IP 172.31.29.11. They have run the following commands:

```
[root@ip-172-31-29-11 temp]# ls
index.html
[root@ip-172-31-29-11 temp]# wget https://wadbucket123.s3.ap-southeast-1.amazonaws.com/main.js
--2023-05-11 10:42:24-- https://wadbucket123.s3.ap-southeast-1.amazonaws.com/main.js
Resolving wadbucket123.s3.ap-southeast-1.amazonaws.com (wadbucket123.s3.ap-southeast-1.amazonaws.com) ... 52.219.124.71
Connecting to wadbucket123.s3.ap-southeast-1.amazonaws.com (wadbucket123.s3.ap-southeast-1.amazonaws.com) [52.219.124.71]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 2004 (2.0K) [application/javascript]
Saving to: 'main.js'

main.js                                     100%[=====] 1.96K --.-KB/s  in 0s

2023-05-11 10:42:24 (27.7 MB/s) - 'main.js' saved [2004/2004]

[root@ip-172-31-29-11 temp]# wget https://wadbucket123.s3.ap-southeast-1.amazonaws.com/style.css
--2023-05-11 10:42:46-- https://wadbucket123.s3.ap-southeast-1.amazonaws.com/style.css
Resolving wadbucket123.s3.ap-southeast-1.amazonaws.com (wadbucket123.s3.ap-southeast-1.amazonaws.com) ... 52.219.128.31
Connecting to wadbucket123.s3.ap-southeast-1.amazonaws.com (wadbucket123.s3.ap-southeast-1.amazonaws.com) [52.219.128.31]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 5480 (5.4K) [text/css]
Saving to: 'style.css'

style.css                                   100%[=====] 5.35K --.-KB/s  in 0s

2023-05-11 10:42:46 (155 MB/s) - 'style.css' saved [5480/5480]

[root@ip-172-31-29-11 temp]# ls
```

A popup window shows the instance ID: i-0375337c0292acb9f (WAD) with Public IP 54.255.207.134 and Private IP 172.31.29.11.

Copy all the files into html directory and start the Apache server:

The screenshot shows the same terminal window where the user has configured and started the Apache HTTP service. The commands and output are as follows:

```
[root@ip-172-31-29-11 html]# systemctl enable httpd
Created symlink /etc/systemd/system/multi-user.target.wants/httpd.service -> /usr/lib/systemd/system/httpd.service.
[root@ip-172-31-29-11 html]# systemctl start httpd
[root@ip-172-31-29-11 html]# systemctl status httpd
```

The status output for the `httpd.service` is shown below:

```
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; enabled; preset: disabled)
   Active: active (running) since Thu 2023-05-11 10:45:59 UTC; 9s ago
     Docs: man:httpd.service(8)
  Main PID: 3509 (httpd)
    Status: "Total requests: 0; Idle/Busy workers 100/0; Requests/sec: 0; Bytes served/sec: 0 B/sec"
    Tasks: 177 (limit: 1108)
   Memory: 17.5M
      CPU: 82ms
  CGroup: /system.slice/httpd.service
          └─3509 /usr/sbin/httpd -DFOREGROUND
             └─3510 /usr/sbin/httpd -DFOREGROUND
                └─3511 /usr/sbin/httpd -DFOREGROUND
                   └─3512 /usr/sbin/httpd -DFOREGROUND
                      └─3513 /usr/sbin/httpd -DFOREGROUND
```

Log messages at the bottom of the terminal confirm the service start:

```
May 11 10:45:59 ip-172-31-29-11.ap-southeast-1.compute.internal systemd[1]: Starting httpd.service - The Apache HTTP Server...
May 11 10:45:59 ip-172-31-29-11.ap-southeast-1.compute.internal systemd[1]: Started httpd.service - The Apache HTTP Server.
May 11 10:45:59 ip-172-31-29-11.ap-southeast-1.compute.internal httpd[3509]: Server configured, listening on: port 80

[root@ip-172-31-29-11 html]#
```

The network info popup is also present, showing the same instance details.

Copy the IPv4 address of EC2 Instance and paste it on browser to see the application running:

The screenshot displays the AWS Management Console interface. On the left, the navigation menu includes options like EC2 Dashboard, EC2 Global View, Events, Limits, and Instances. The main panel shows a list of EC2 instances with columns for Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, and Public IPv4 DNS. One instance, 'WAD' with ID 'i-0375337c0292acb9f', is shown in a 'Running' state. Below the list, the details for this instance are expanded, showing tabs for Details, Security, Networking, Storage, Status checks, Monitoring, and Tags. The 'Details' tab is active, displaying the Instance ID, IPv6 address, Hostname type, and a 'Public IPv4 address copied' notification. The instance is running on a 't2.micro' instance type in the 'ap-southeast-1' region. The bottom of the console shows the 'CloudShell' tab and a file explorer with two image files.

Hence, we deployed a static website using EC2 and S3 bucket:

The screenshot shows a web browser displaying a static website. The website has a dark brown header with the title 'Enrollment Form'. Below the header, there are two main sections: 'Enroll Now' and 'Registered Candidates'. The 'Enroll Now' section contains a form with fields for Name, Email, Your Photo, Birth Date, and Gender. The 'Registered Candidates' section displays the details of a registered candidate, Harshita Totala, including her name, email, DOB, gender, and a photo of a pink flower. The browser's address bar shows the URL '54.255.207.134'.