

Assignment - AWS Overview

Q1)List out the types of instance base on the pricing model and write a brief about your understanding about it.

Ans -

Instance Pricing Model allows us to choose the most cost-effective and efficient way to run your workloads based on usage patterns. Here are the main types of instance pricing models:

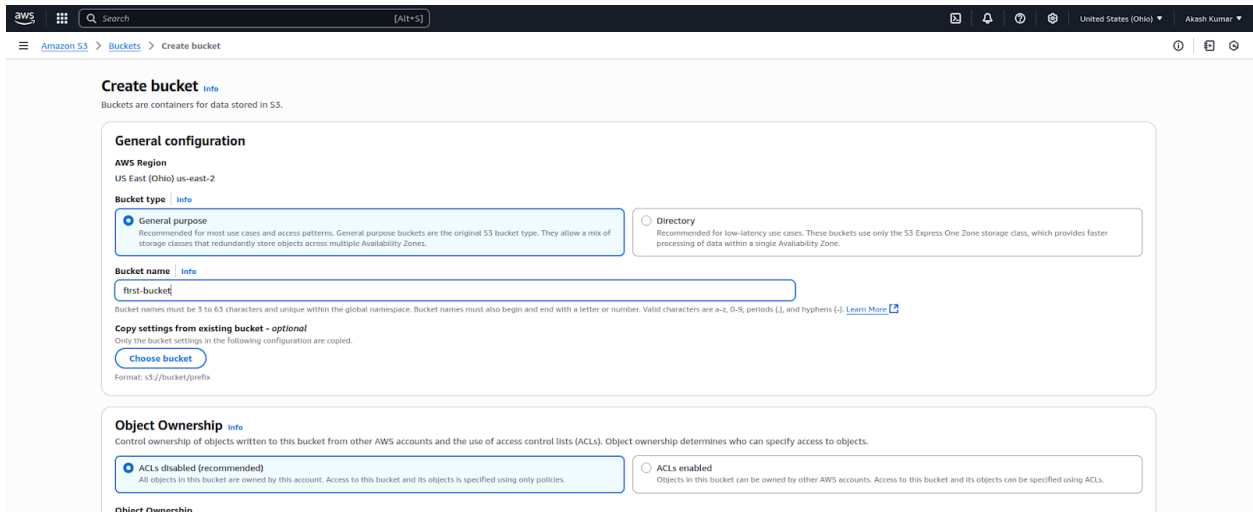
| <u>Instance Type</u> | <u>Description</u> | <u>Best for</u> | <u>Commitment</u> | <u>Cost Saving</u> | <u>Flexibility</u> |
|-----------------------|--|---|------------------------------------|---------------------------------|--------------------|
| On-Demand | pay-per-hour | Short term, Unpredictable workloads | 0 | low | High |
| Reserved Instances | Pay for 1 or 3 years for a discount | Predictable, Steady workloads | 1 or 3 years | Upto 75% | Low to Medium |
| Savings Plans | Commit to a consistent per-hour usage for 1 or 3 years for lower rates | Flexible but Predictable workloads | 1 or 3 years | Upto 72% | High |
| Spot Instances | Bid for unused EC2 capacity, can be interrupted | Fault Tolerant, flexible applications | 0 | Upto 90% | Medium |
| Dedicated Host | Physical server dedicated to your account | Compliance , licensing, and workload isolation | optional(on demand or reservation) | Moderate to high | Low to Medium |
| Dedicated Instances | Run instances on hardware dedicated to you without full host control | Security/isolation needs without licensing concerns | 0 | Slightly higher than on demand | Medium |
| Capacity Reservations | Reserve capacity in a specific AZ without | Capacity-critical workloads, disaster | flexible(can cancel anytime) | No cost savings on usage itself | High |

| | | | | | |
|--|------------------------------|----------|--|--|--|
| | pre-paying for instance time | recovery | | | |
|--|------------------------------|----------|--|--|--|

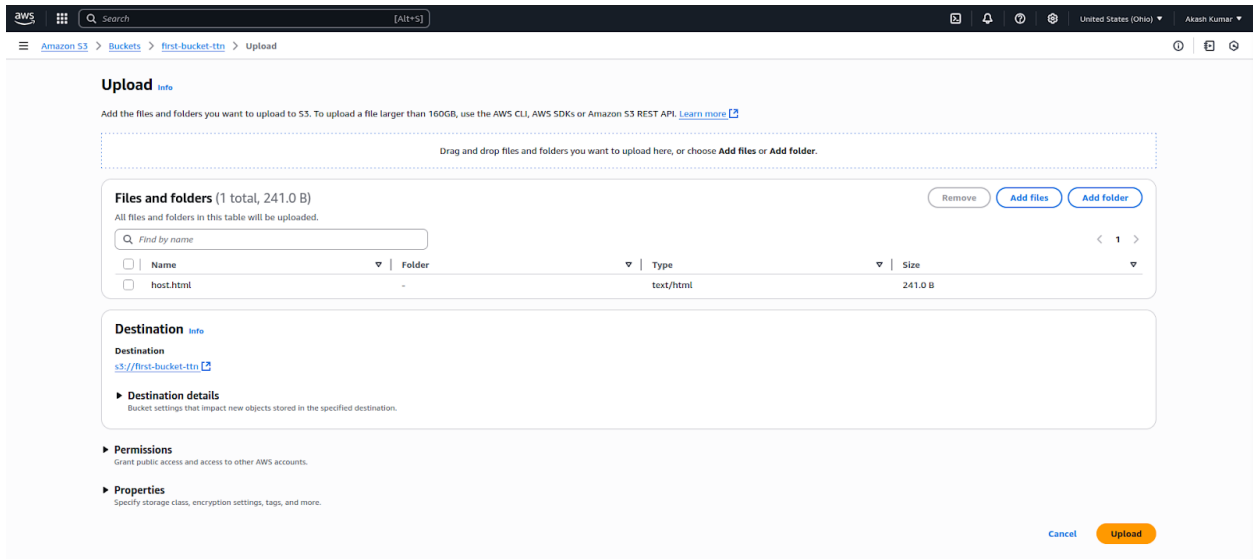
Q2)Host a static website in S3.

Ans - Steps Involved -

- 1) Search S3
- 2) Then in S3 , Click Create bucket.



3) Now upload your development files and folders



4) Now go to Bucket and provide certain permissions listed below

- Goto properties -> allow static site hosting
- Next goto permission -> allow public access and enable ACL(access control list)
- Now go to objects and select all the objects you want to give public access->goto action -> goto make public using ACL.

Upload: status

After you navigate away from this page, the following information is no longer available.

Summary

| Destination | Succeeded | Failed |
|-----------------------|---------------------------|-------------------|
| s3://first-bucket-ttn | 1 file, 241.0 B (100.00%) | 0 files, 0 B (0%) |

Files and folders (1 total, 241.0 B)

| Name | Folder | Type | Size | Status | Error |
|-----------|--------|-----------|---------|-----------|-------|
| host.html | - | text/html | 241.0 B | Succeeded | - |

5) Now click on object url

host.html

Object overview

| | |
|--|--|
| Owner 57a1432126de07cc52bc40c5aa1b11b219017e05b904b2b9625e3da8aac7576b | S3 URI s3://first-bucket-ttn/host.html |
| AWS Region US East (Ohio) us-east-2 | Amazon Resource Name (ARN) arn:aws:s3::first-bucket-ttn/host.html |
| Last modified June 17, 2023, 14:17:17 (UTC+05:30) | Entity tag (ETag) c560066ec08c9fc54855c8fc66e9086b |
| Size 241.0 B | Object URL https://first-bucket-ttn.s3.us-east-2.amazonaws.com/host.html |
| Type html | |
| Key host.html | |

Object management overview

The following bucket properties and object management configurations impact the behavior of this object.

Bucket properties

Bucket Versioning

When enabled, multiple variants of an object can be stored in the bucket to easily recover from unintended user actions and application failures.

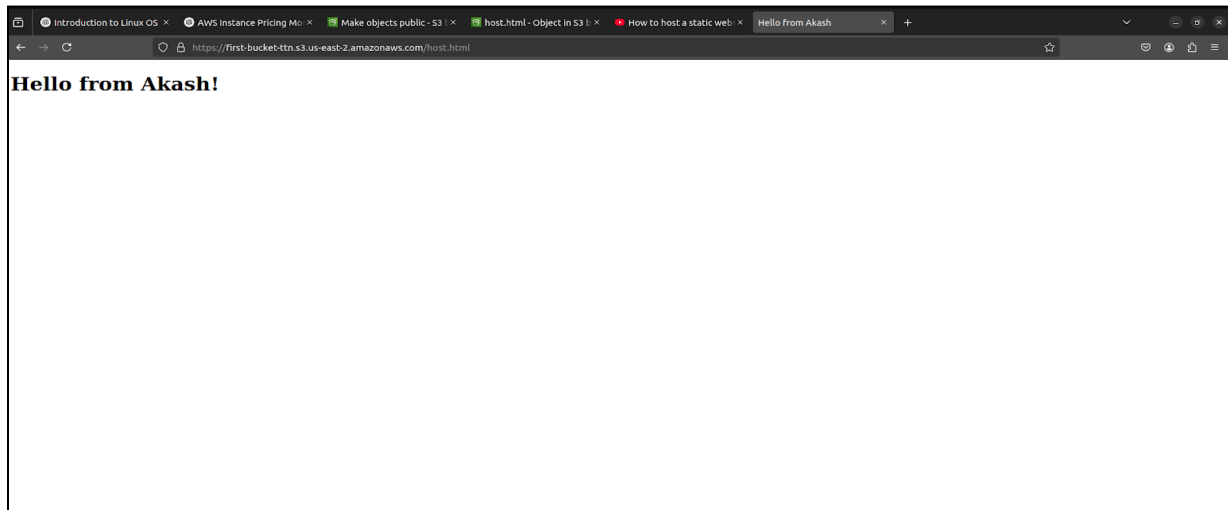
Disabled

Management configurations

Replication status

When a replication rule is applied to an object the replication status indicates the progress of the operation.

6) Site hosted successfully

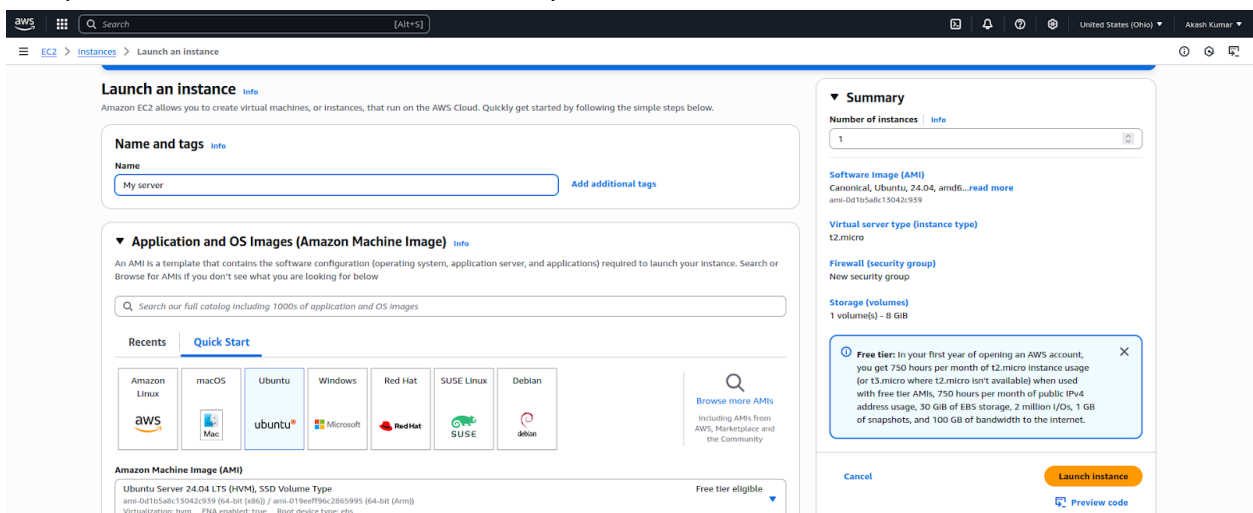


Q3) Launch an Ubuntu EC2 instance on AWS, with 10GB root volume, and SSH from your local machine using the private key.

Ans -

Steps Involved -

- 1) Goto EC2 -> Instances
- 2) Click on launch instance and fill required details



3) Generate Key Pair (Pem Key)

On-Demand Linux base pricing: \$0.091 per hour per instance
On-Demand Linux base pricing: \$0.091 per hour per instance
On-Demand RHEL base pricing: \$0.026 USD per Hour
On-Demand Windows base pricing: \$0.0162 USD per Hour

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

Select [Create new key pair](#)

Network settings [Info](#) [Edit](#)

Network [Info](#)
vpc-0655bbec0a489d67e

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

Summary

Number of instances [Info](#)
1

Software image (AMI)
Canonical, Ubuntu, 24.04, amd64... [read more](#)
ami-0d1b5a1c13042c939

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 of opening an AWS account, you get 750 hours per month of t2.micro instance usage (or t3.micro where t2.micro isn't available) when used with free tier AMIs, 750 hours per month of public IPv4 address usage, 30 GiB of EBS storage, 2 million I/Os, 1 GiB of snapshots, and 100 GiB of bandwidth to the Internet.

[Cancel](#) [Launch instance](#) [Preview code](#)

4) EC2 instance created successfully

Activities [Firefox Web Browser](#) Jun 16 14:49

Bootcamp Portal [Launch an Instance | EC2](#) [EC2 setup guide](#)

[https://us-east-2.console.aws.amazon.com/ec2/home?region=us-east-2#LaunchInstances:](#)

Success
Successfully initiated launch of instance (i-0e5db42ba24724f39)

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.
[Create billing alerts](#)

Connect to your instance
Once your instance is running, log into it from your local computer.
[Connect to instance](#)
[Learn more](#)

Connect an RDS database
Configure the connection between an EC2 instance and a database to allow traffic flow between them.
[Connect an RDS database](#)
[Create a new RDS database](#)
[Learn more](#)

Create EBS snapshot policy
Create a policy that automates the creation, retention, and deletion of EBS snapshots
[Create EBS snapshot policy](#)

Manage detailed monitoring
Enable or disable detailed monitoring for the instance. If you enable detailed monitoring, the Amazon EC2 console displays monitoring graphs with a 1-minute period.
[Manage detailed monitoring](#)

Create Load Balancer
Create an application, network gateway or classic Elastic Load Balancer
[Create Load Balancer](#)

Create AWS budget
AWS Budgets allows you to create budgets, forecast spend, and take action on your costs and usage from a single location.
[Create AWS budget](#)

Manage CloudWatch alarms
Create or update Amazon CloudWatch alarms for the instance.
[Manage CloudWatch alarms](#)

Disaster recovery for your instances
Recover the instances you just launched into a different Availability Zone or a different Region using AWS Elastic Disaster Recovery (DRS).
[Disaster recovery for your](#)

Monitor for suspicious runtime activities
Amazon GuardDuty enables you to continuously monitor for malicious runtime activity and unauthorized behavior, with near real-time visibility into on-host activities occurring across your Amazon EC2 workloads.
[Get instance screenshot](#)

Get instance screenshot
Capture a screenshot from the instance and view it as an image. This is useful for troubleshooting an unreachable instance.
[Get instance screenshot](#)

Get system log
View the instance's system log to troubleshoot issues.
[Get system log](#)

© 2025, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

5) Now you can see all the available instances

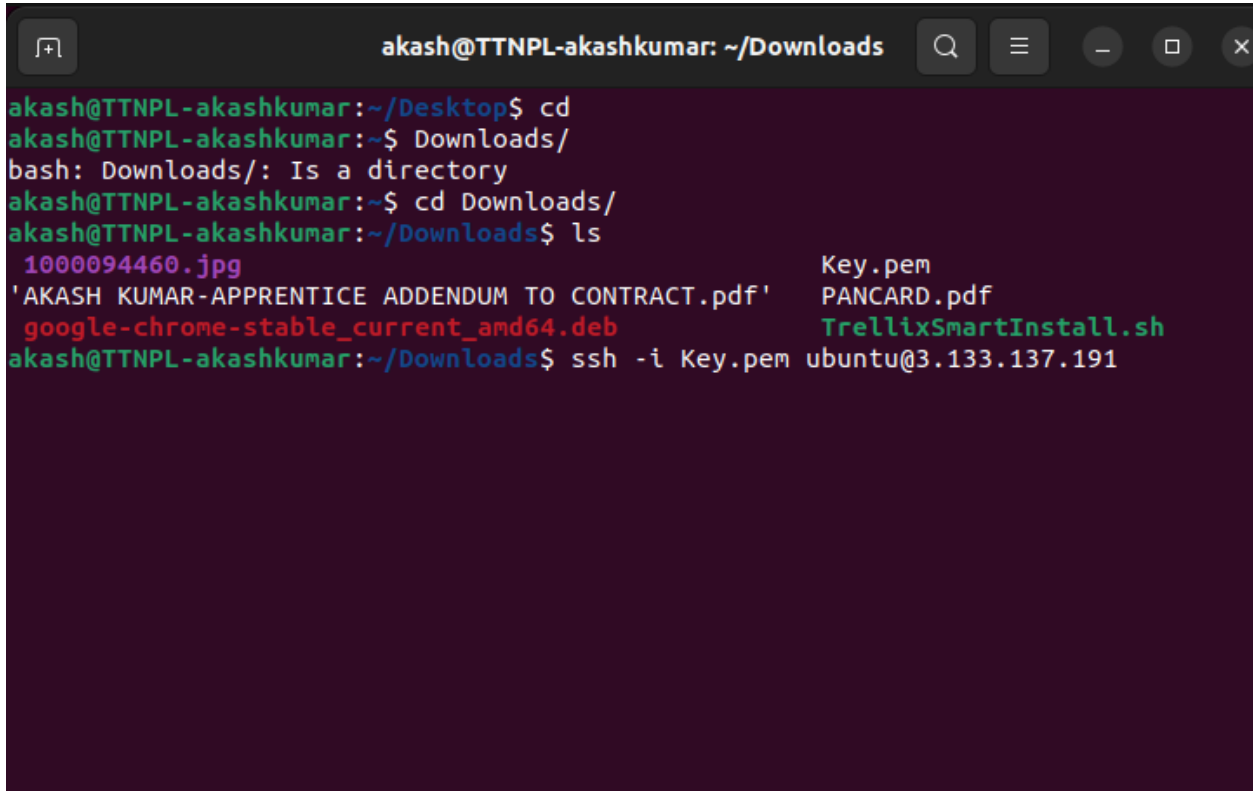
The screenshot displays the AWS Management Console for the EC2 service. The left-hand navigation pane shows the 'Instances' section selected. The main content area, titled 'Instances (1)', shows a table with one instance: 'My Web Server'. The instance is in a 'Running' state, using the 't2.micro' instance type. The table columns include Name, Instance ID, Instance state, Instance type, Status check, Alarm status, Availability Zone, Public IP-v4 DNS, Public IP-v4, and Elastic IP. Below the table, there is a 'Select an instance' section. The bottom of the screen shows the Linux desktop environment with various application icons in the dock.

| Name | Instance ID | Instance state | Instance type | Status check | Alarm status | Availability Zone | Public IP-v4 DNS | Public IP-v4 | Elastic IP |
|---------------|---------------------|----------------|---------------|--------------|---------------|-------------------|-----------------------|---------------|------------|
| My Web Server | i-0e5db42ba24724f39 | Running | t2.micro | Initializing | View alarms + | us-east-2a | ec2-53-137-191.us-... | 5.133.137.191 | - |

Q4) Install nginx package in the above server and access this page from your local browser using a domain name instead of IP address of the server.

Ans- Steps Involved

- 1) Goto folder where you pem key is stored
- 2) Then run command -
 - `sudo ssh -i pem_key_name.pem system@public_ip_of_instances`

A terminal window titled 'akash@TTNPL-akashkumar: ~/Downloads' with standard window controls. The terminal shows a sequence of commands: 'cd' (from ~/Desktop to ~), 'Downloads/' (resulting in an error), 'cd Downloads/', and 'ls' (listing files like 1000094460.jpg, Key.pem, PANCARD.pdf, etc.). The final command is 'ssh -i Key.pem ubuntu@3.133.137.191'.

```
akash@TTNPL-akashkumar:~/Desktop$ cd
akash@TTNPL-akashkumar:~$ Downloads/
bash: Downloads/: Is a directory
akash@TTNPL-akashkumar:~$ cd Downloads/
akash@TTNPL-akashkumar:~/Downloads$ ls
1000094460.jpg                               Key.pem
'AKASH KUMAR-APPRENTICE ADDENDUM TO CONTRACT.pdf' PANCARD.pdf
google-chrome-stable_current_amd64.deb      TrellixSmartInstall.sh
akash@TTNPL-akashkumar:~/Downloads$ ssh -i Key.pem ubuntu@3.133.137.191
```

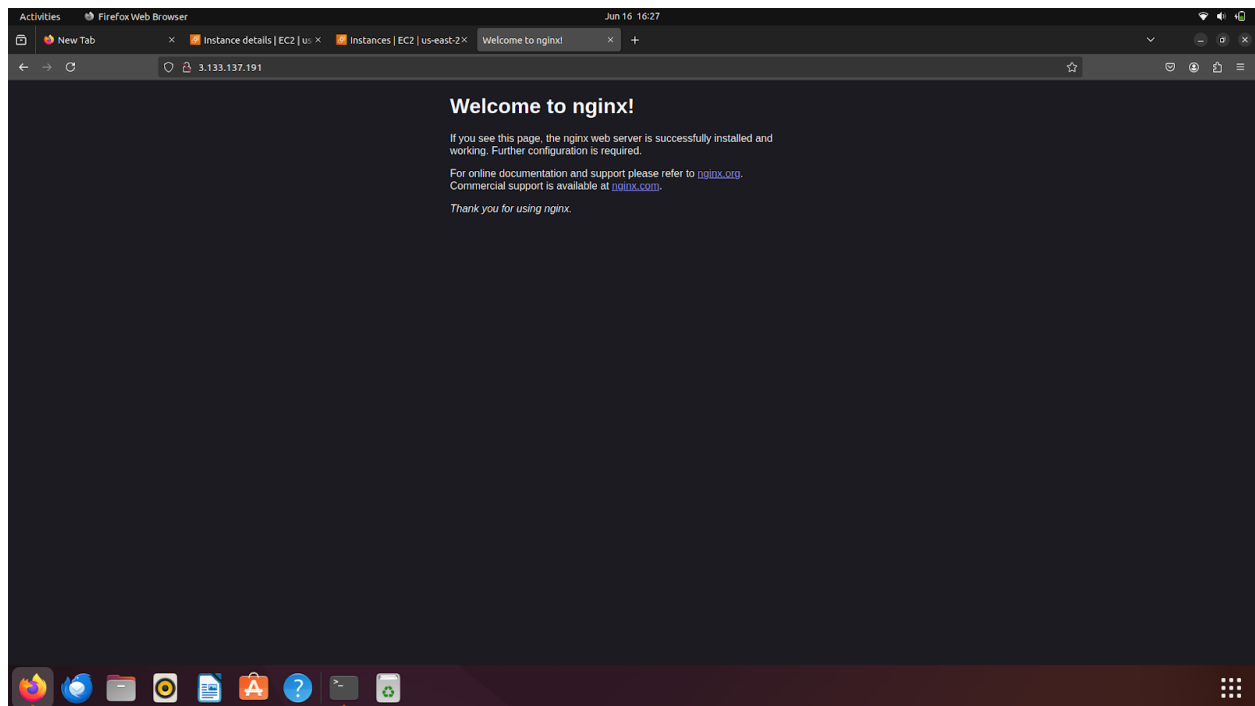
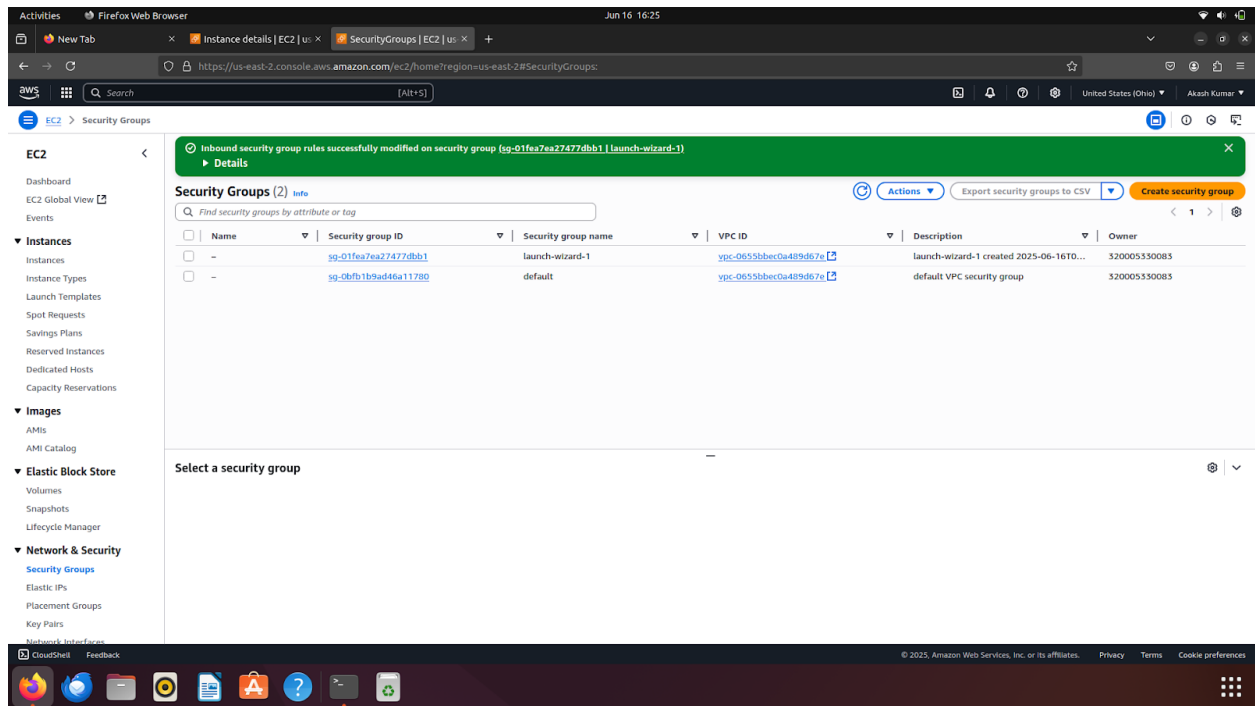
```
ubuntu@ip-172-31-15-95: ~  
It is required that your private key files are NOT accessible by others.  
This private key will be ignored.  
Load key "Key.pem": bad permissions  
ubuntu@3.133.137.191: Permission denied (publickey).  
akash@TTNPL-akashkumar:~/Downloads$ sudo ssh -i Key.pem ubuntu@3.133.137.191  
[sudo] password for akash:  
The authenticity of host '3.133.137.191 (3.133.137.191)' can't be established.  
ED25519 key fingerprint is SHA256:TlpD8LrP7XzYjL95eC3msckfAvwMa3UcgDRmv2Lm6ME.  
This key is not known by any other names  
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes  
Warning: Permanently added '3.133.137.191' (ED25519) to the list of known hosts.  
Welcome to Ubuntu 24.04.2 LTS (GNU/Linux 6.8.0-1029-aws x86_64)  
  
* Documentation:  https://help.ubuntu.com  
* Management:    https://landscape.canonical.com  
* Support:        https://ubuntu.com/pro  
  
System information as of Mon Jun 16 09:23:39 UTC 2025  
  
System load:  0.0           Processes:      107  
Usage of /:   19.7% of 8.65GB Users logged in: 0  
Memory usage: 20%          IPv4 address for enX0: 172.31.15.95  
Swap usage:   0%  
  
Expanded Security Maintenance for Applications is not enabled.  
  
0 updates can be applied immediately.  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
The programs included with the Ubuntu system are free software;  
the exact distribution terms for each program are described in the  
individual files in /usr/share/doc/*/copyright.  
  
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by  
applicable law.  
  
To run a command as administrator (user "root"), use "sudo <command>".  
See "man sudo_root" for details.  
ubuntu@ip-172-31-15-95:~$
```


3) Now install nginx using command

- sudo apt update
- sudo apt install nginx

```
ubuntu@ip-172-31-15-95: ~  
Building dependency tree... Done  
Reading state information... Done  
1 package can be upgraded. Run 'apt list --upgradable' to see it.  
ubuntu@ip-172-31-15-95:~$ sudo apt install nginx  
Reading package lists... Done  
Building dependency tree... Done  
Reading state information... Done  
The following additional packages will be installed:  
  nginx-common  
Suggested packages:  
  fcgiwrap nginx-doc ssl-cert  
The following NEW packages will be installed:  
  nginx nginx-common  
0 upgraded, 2 newly installed, 0 to remove and 1 not upgraded.  
Need to get 551 kB of archives.  
After this operation, 1596 kB of additional disk space will be used.  
Do you want to continue? [Y/n] Y  
Get:1 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 nginx-common all 1.24.0-2ubuntu7 [31.2 kB]  
Get:2 http://us-east-2.ec2.archive.ubuntu.com/ubuntu noble-updates/main amd64 nginx amd64 1.24.0-2ubuntu7.3 [530 kB]  
Fetched 551 kB in 0s (16.2 MB/s)  
Preconfiguring packages ...  
Selecting previously unselected package nginx-common.  
(Reading database ... 70681 files and directories currently installed.)  
Preparing to unpack .../nginx-common_1.24.0-2ubuntu7.3_all.deb ...  
Unpacking nginx-common (1.24.0-2ubuntu7.3) ...  
Selecting previously unselected package nginx.  
Preparing to unpack .../nginx_1.24.0-2ubuntu7.3_amd64.deb ...  
Unpacking nginx (1.24.0-2ubuntu7.3) ...  
Setting up nginx (1.24.0-2ubuntu7.3) ...  
Setting up nginx-common (1.24.0-2ubuntu7.3) ...  
Created symlink /etc/systemd/system/multi-user.target.wants/nginx.service → /usr/lib/systemd/system/nginx.service.  
Processing triggers for ufw (0.36.2-6) ...  
Processing triggers for man-db (2.12.0-4build2) ...  
Scanning processes...  
Scanning linux images...  
  
Running kernel seems to be up-to-date.
```

- 4) Creating inbound rule to expose port 80
- 5) Using duckdns to get free domain name



6) Now we need to edit the nginx config file to add the domain name

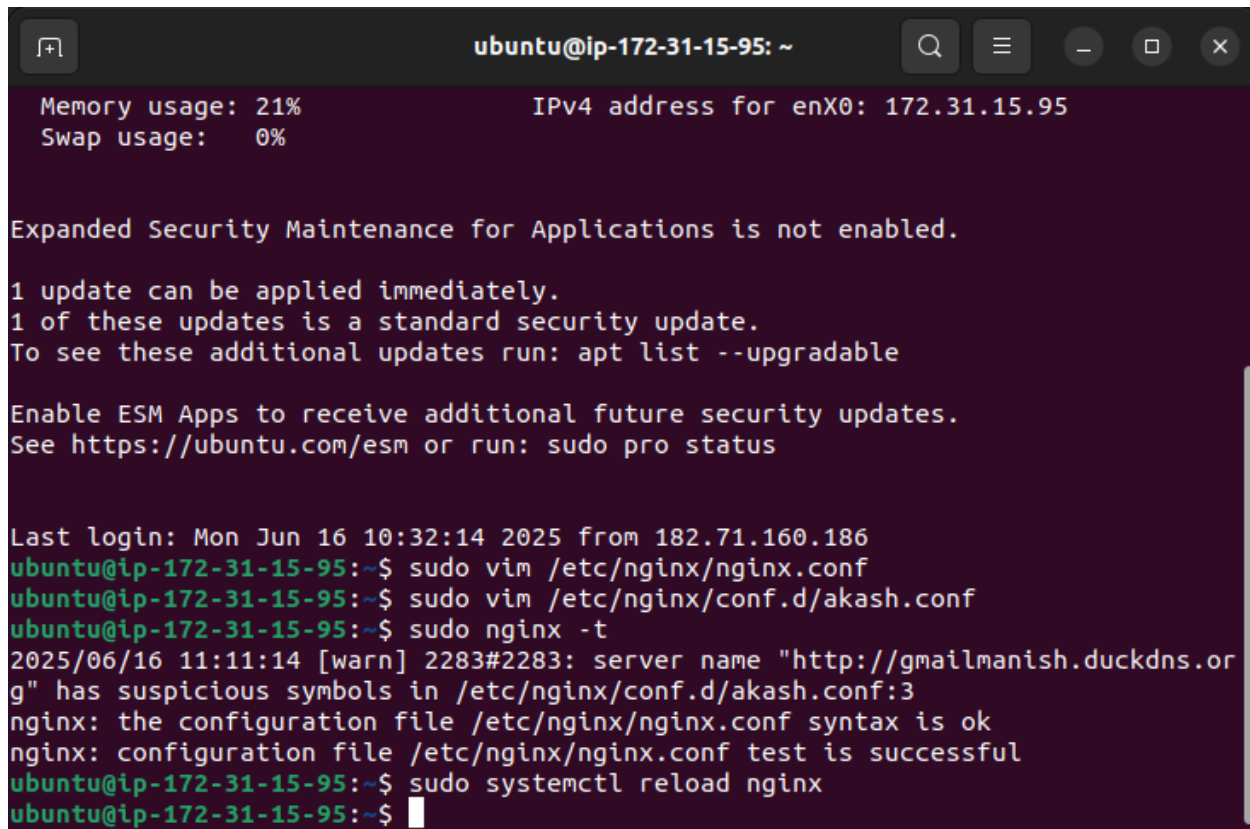
7) Add following details in config file -

```
- server {  
    listen 80;  
    server_name <your_domain_on_duckdns>;  
    location / {  
        root /usr/share/nginx/html;  
        Index index.html  
    }  
}
```

What above configuration do

- DNS resolves it to your server's public IP (set in DuckDNS)
- NGINX receives the request on port 80.
- This config matches the hostname
- It serves the index.html file from /usr/share/nginx/html.

8) Now we reload the nginx using `sudo systemctl reload nginx`



```
ubuntu@ip-172-31-15-95: ~  
Memory usage: 21%          IPv4 address for enX0: 172.31.15.95  
Swap usage: 0%  
  
Expanded Security Maintenance for Applications is not enabled.  
1 update can be applied immediately.  
1 of these updates is a standard security update.  
To see these additional updates run: apt list --upgradable  
  
Enable ESM Apps to receive additional future security updates.  
See https://ubuntu.com/esm or run: sudo pro status  
  
Last login: Mon Jun 16 10:32:14 2025 from 182.71.160.186  
ubuntu@ip-172-31-15-95:~$ sudo vim /etc/nginx/nginx.conf  
ubuntu@ip-172-31-15-95:~$ sudo vim /etc/nginx/conf.d/akash.conf  
ubuntu@ip-172-31-15-95:~$ sudo nginx -t  
2025/06/16 11:11:14 [warn] 2283#2283: server name "http://gmailmanish.duckdns.org" has suspicious symbols in /etc/nginx/conf.d/akash.conf:3  
nginx: the configuration file /etc/nginx/nginx.conf syntax is ok  
nginx: configuration file /etc/nginx/nginx.conf test is successful  
ubuntu@ip-172-31-15-95:~$ sudo systemctl reload nginx  
ubuntu@ip-172-31-15-95:~$
```

