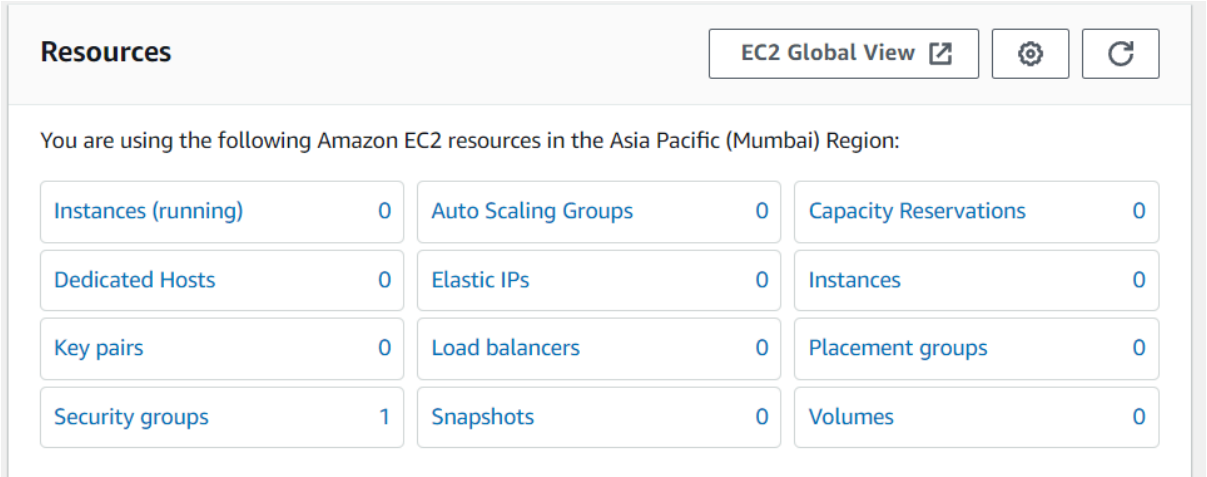
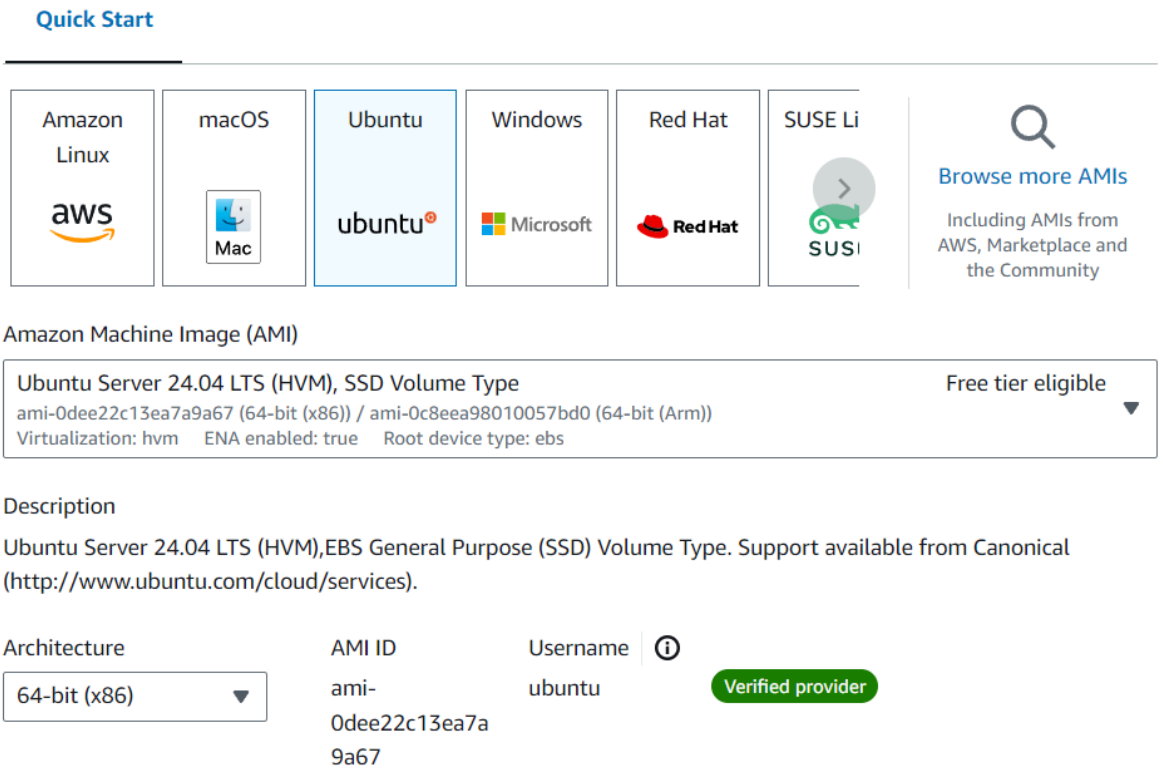
AWS -EC2
Elastic Compute Cloud

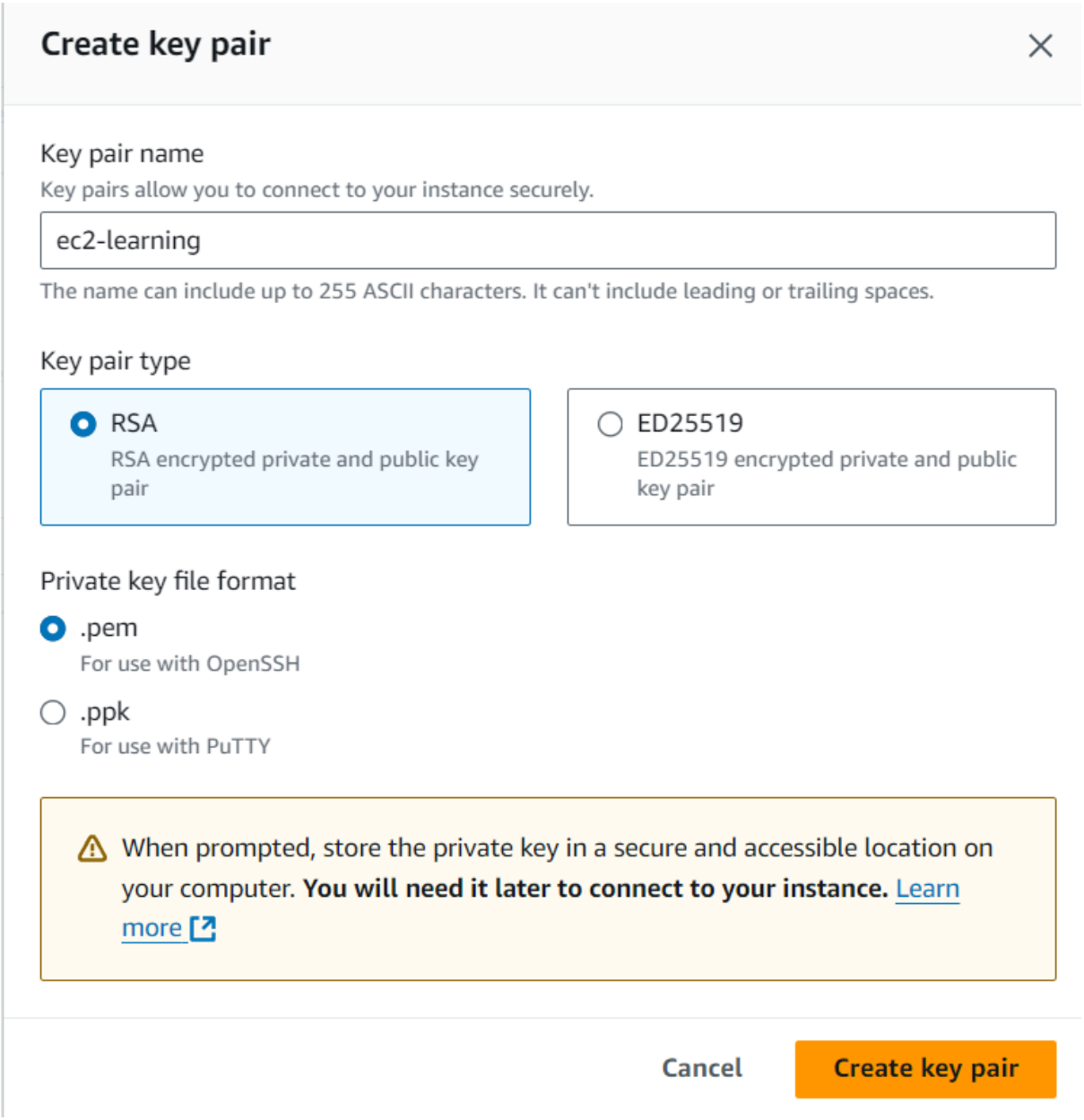
Dashboard:



Snapshots:

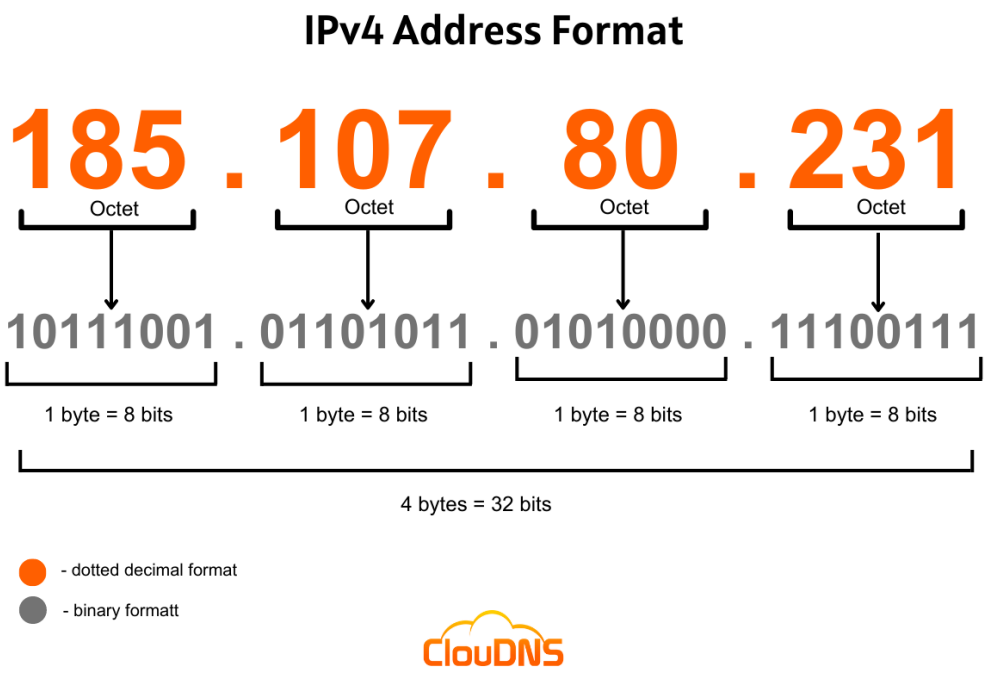


Create a Key-Pair 👍

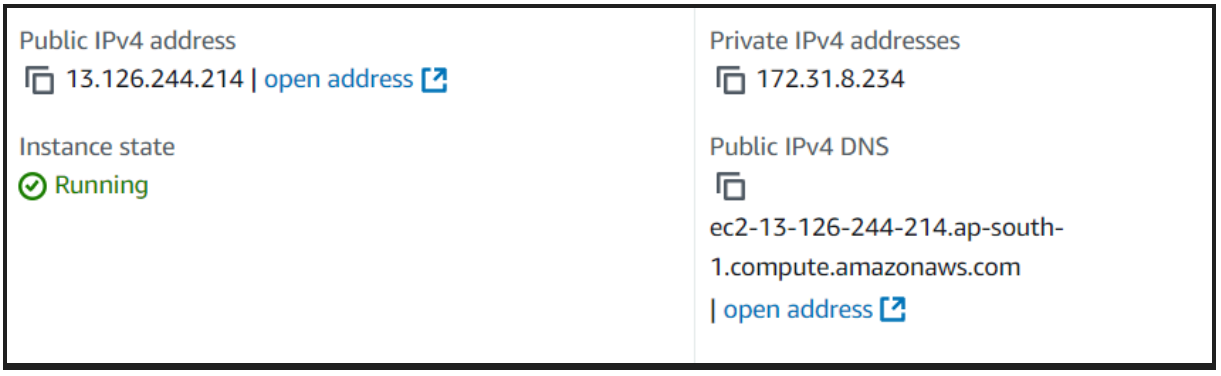


An IPv4 address is a 32-bit number that uniquely identifies a network interface on a machine.

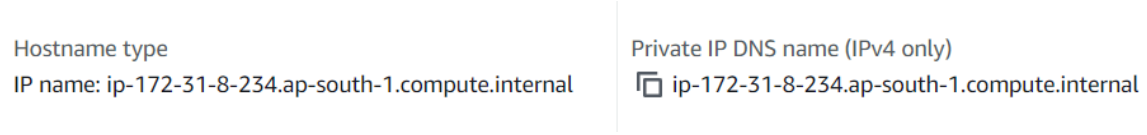
IPv4 addresses are made up of two parts:
Network prefix: Specifies the unique number assigned to a network
Host number: Uniquely identifies a host within a network



An IPv6 address is a 128-bit alphanumeric value that identifies a device in an Internet Protocol Version 6 (IPv6) network. IPv6 is an advanced networking standard that allows for a much larger number of unique IP addresses than the older IPv4 standard.



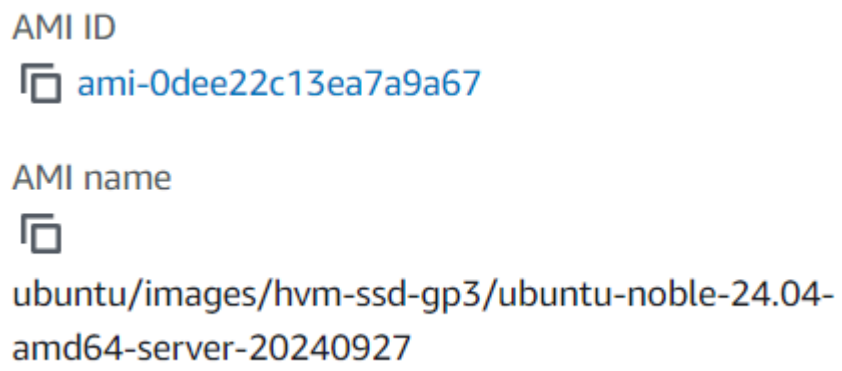
Difference between Public and Private IPv4 address:
Public IPv4 addresses are routable on the internet and assigned by IANA, allowing direct access from any device online.
Private IPv4 addresses are used within local networks and are not routable on the internet, falling within specific ranges (e.g., 192.168.x.x).



- Public Hostname: Accessible from the internet (e.g., `ec2-203-0-113-25.compute-1.amazonaws.com`).
- Private Hostname: Accessible only within the VPC (e.g., `ip-10-0-1-23.ec2.internal`).
- Private IP DNS Name: Resolves to the private IP (e.g., `ip-10-0-1-23.ec2.internal`).

What's VPC?
A VPC (Virtual Private Cloud) is a virtual network dedicated to your AWS account. It allows you to launch AWS resources in a logically isolated environment.
You can define your own network configuration, including IP address ranges, subnets, route tables, and network gateways.

AMI:



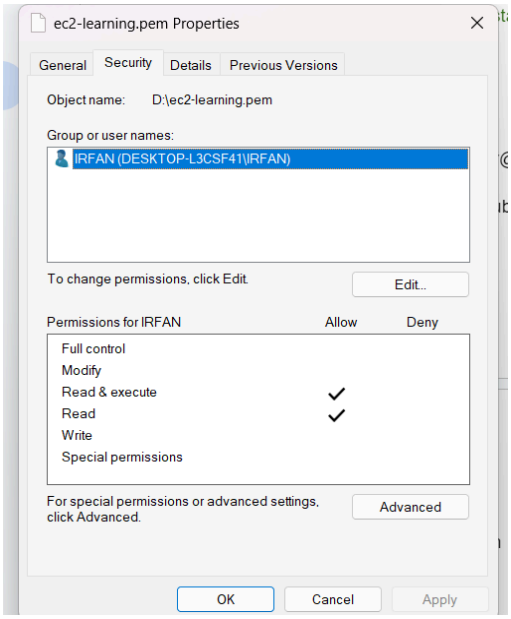
An AMI (Amazon Machine Image) is a pre-configured template used to create virtual machines (EC2 instances) on AWS. It contains the operating system, application server, and applications required to launch an instance.

Connection:

Command:
ssh -i /path/to/your-key.pem ec2-user@your-ec2-public-dns

- @your-ec2-public-dns: The public DNS or IP address of your EC2 instance.

Changing Permissions:
chmod 400 /path/to/ec2-learning.pem
On Windows:



Some Commands:

- cat /etc/os-release

```
ubuntu@ip-172-31-8-234:~$ cat /etc/os-release
PRETTY_NAME="Ubuntu 24.04.1 LTS"
NAME="Ubuntu"
VERSION_ID="24.04"
VERSION="24.04.1 LTS (Noble Numbat)"
VERSION_CODENAME=noble
ID=ubuntu
ID_LIKE=debian
HOME_URL="https://www.ubuntu.com/"
SUPPORT_URL="https://help.ubuntu.com/"
BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/"
PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy"
UBUNTU_CODENAME=noble
LOGO=ubuntu-logo
```

Know CPU Details:

```
ubuntu@ip-172-31-8-234:~$ lscpu
Architecture:             x86_64
  CPU op-mode(s):         32-bit, 64-bit
  Address sizes:          46 bits physical,
                          48 bits virtual
  Byte Order:             Little Endian
CPU(s):                   1
  On-line CPU(s) list:    0
Vendor ID:                GenuineIntel
  Model name:             Intel(R) Xeon(R) C
                          PU E5-2686 v4 @ 2.
                          30GHz
    CPU family:           6
```

The command df -h is used in Unix/Linux systems to display information about disk space usage in a human-readable format.

```
ubuntu@ip-172-31-8-234:~$ df -h
Filesystem      Size  Used Avail Use% Mounte
d on
/dev/root       6.8G  1.6G  5.2G  24% /
tmpfs           479M     0  479M   0% /dev/s
hm
tmpfs           192M  884K  191M   1% /run
tmpfs           5.0M     0  5.0M   0% /run/l
ock
/dev/xvda16     881M   76M  744M  10% /boot
/dev/xvda15     105M  6.1M   99M   6% /boot/
efi
tmpfs            96M   12K   96M   1% /run/u
ser/1000
```

INSTALLING NGINX ⚒️

1. sudo -i : it opens a new shell with root user privileges, providing a login environment as the root user.
2. apt-get update : refreshes the package list for upgrades on Debian-based systems.
3. apt-get install nginx : installs the Nginx web server on Debian-based systems.
4. service nginx status : checks the current status of the Nginx service, indicating whether it is running, stopped, or inactive.

```
root@ip-172-31-8-234:~# service nginx status
● nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/usr/lib/systemd/system/nginx.service; enabled; preset: enabled)
     Active: active (running) since Tue 2024-10-08 10:17:31 UTC; 15s ago
       Docs: man:nginx(8)
    Process: 1944 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 1946 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 1951 (nginx)
      Tasks: 2 (limit: 1130)
     Memory: 1.7M (peak: 1.9M)
        CPU: 7ms
     CGroup: /system.slice/nginx.service
             ├─1951 "nginx: master process /usr/sbin/nginx -g daemon on; master_process on;"
             └─1952 "nginx: worker process"
```

5. curl localhost : sends an HTTP request to the local server (localhost) and displays the response, typically the content served by a web server running on the same machine.

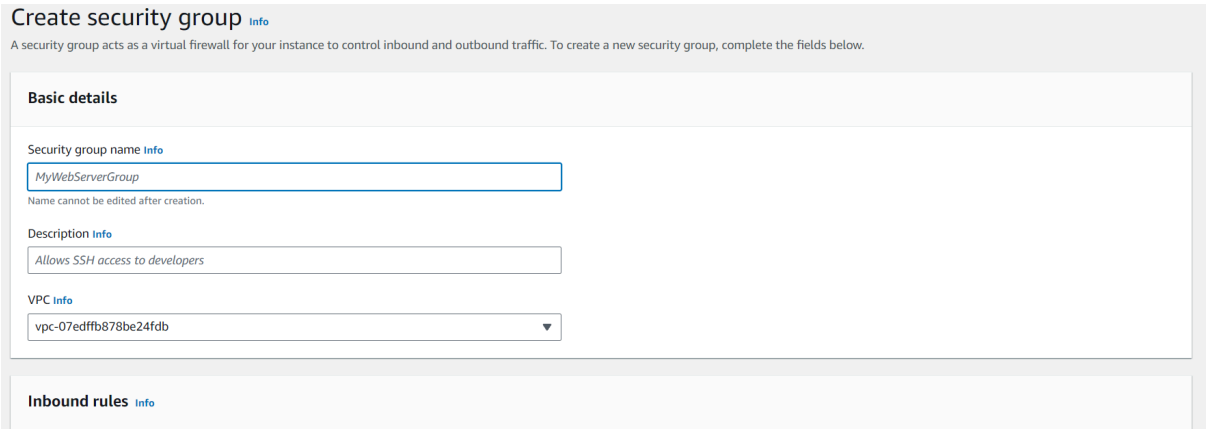
```
root@ip-172-31-8-234:~# curl localhost
<!DOCTYPE html>
<html>
<head>
<title>Welcome to nginx!</title>
<style>
html { color-scheme: light dark; }
body { width: 35em; margin: 0 auto;
font-family: Tahoma, Verdana, Arial, sans-serif; }
</style>
</head>
<body>
<h1>Welcome to nginx!</h1>
<p>If you see this page, the nginx web server is successfully installed and
working. Further configuration is required.</p>

<p>For online documentation and support please refer to
<a href="http://nginx.org/">nginx.org</a>.<br/>
Commercial support is available at
<a href="http://nginx.com/">nginx.com</a>.</p>

<p><em>Thank you for using nginx.</em></p>
</body>
</html>
```

AWS Security Groups act as virtual firewalls for your EC2 instances to control inbound and outbound traffic.
Here's a brief overview:

Key Features:
Stateful: If you allow an incoming request from an IP, the response is automatically allowed, regardless of outbound rules.
Rules: You define rules based on:
Protocol: TCP, UDP, ICMP, etc.
Port Range: Specific ports or ranges (e.g., 22 for SSH, 80 for HTTP).
Source/Destination: IP addresses or CIDR blocks (e.g., 0.0.0.0/0 for all IPs).
Default Security Group: Each VPC has a default security group that allows all outbound traffic and no inbound traffic.
Multiple Security Groups: You can assign multiple security groups to an instance, and the rules are aggregated.
Use Cases:
Allowing SSH access from specific IPs.
Restricting web traffic to certain ports.
Managing access for different application tiers (web, app, database).
Management:
Security groups can be modified at any time, and changes take effect immediately.



The main difference between the HTTP and HTTPS port numbers is that HTTP uses port 80, while HTTPS uses port 443:
Port 22 is used for Secure Shell (SSH) communication,

PLANS:

Summary:

Type	Pricing	Use Case	Flexibility	Commitment
Reserved Instances	Up to 75% savings	Long-term, predictable workloads	Limited flexibility	1-3 years
Spot Instances	Up to 90% savings	Fault-tolerant, interruptible workloads	Highly flexible, but may terminate	None
Savings Plans	Up to 72% savings	Broad workload flexibility (not tied to specific instances)	Very flexible (cross-instance, region, service)	1-3 years
Dedicated Hosts	Pay for entire host server	Workloads needing dedicated hardware	Physical server control	On-Demand or Reserved

Instance Metadata With UserData

curl 169.254.169.254 is used to make an HTTP request to the IP address 169.254.169.254, which is commonly associated with the AWS Instance Metadata Service

Here are the five most important curl commands to retrieve metadata from the AWS EC2 Instance Metadata Service:

Get Instance ID:

Copy
curl http://169.254.169.254/latest/meta-data/instance-id
Get Instance Type:

Copy
curl http://169.254.169.254/latest/meta-data/instance-type
Get Public IP Address:

Copy
curl http://169.254.169.254/latest/meta-data/public-ipv4
Get Private IP Address:

Copy
curl http://169.254.169.254/latest/meta-data/local-ipv4
Get AMI ID:

Copy
curl http://169.254.169.254/latest/meta-data/ami-id
These commands will give you essential information about your EC2 instance.

Attach Elastic/Static IP to an EC2 Instance:

Click the Elastic IPs link in the EC2 Dashboard.
Click Allocate New Address and choose VPC or EC2 from the drop-down list, depending whether you're going to associate this IP with an instance in Amazon EC2-Virtual Private Cloud (VPC) or Amazon EC2-Classic, respectively.
Click Yes, Allocate to confirm your choice.