**Amazon EC2**

**What is Amazon EC2?**

Amazon Elastic Compute Cloud (Amazon EC2) provides on-demand, scalable computing capacity in the Amazon Web Services (AWS) Cloud. Using Amazon EC2 reduces hardware costs so you can develop and deploy applications faster. You can use Amazon EC2 to launch as many or as few virtual servers as you need, configure security and networking, and manage storage. You can add capacity (scale up) to handle compute-heavy tasks, such as monthly or yearly processes, or spikes in website traffic. When usage decreases, you can reduce capacity (scale down) again.

The following diagram shows a basic architecture of an Amazon EC2 instance deployed within an Amazon Virtual Private Cloud (VPC). In this example, the EC2 instance is within an Availability Zone in the Region. The EC2 instance is secured with a security group, which is a virtual firewall that controls incoming and outgoing traffic. A private key is stored on the local computer and a public key is stored on the instance. Both keys are specified as a key pair to prove the identity of the user. In this scenario, the instance is backed by an Amazon EBS volume. The VPC communicates with the internet using an internet gateway. For more information about Amazon VPC,

**
   A basic architecture diagram of an EC2 instance within a VPC.
  **

**Features of Amazon EC2:**

Amazon EC2 provides the following high-level features:

**Instances**

Virtual servers.

**Amazon Machine Images (AMIs)**

Preconfigured templates for your instances that package the components you need for your server (including the operating system and additional software).

**Instance types**

Various configurations of CPU, memory, storage, networking capacity, and graphics hardware for your instances.

**Key pairs**

Secure login information for your instances. AWS stores the public key and you store the private key in a secure place.

**Instance store volumes**

Storage volumes for temporary data that is deleted when you stop, hibernate, or terminate your instance.

**Amazon EBS volumes**

Persistent storage volumes for your data using Amazon Elastic Block Store (Amazon EBS).

**Regions, Availability Zones, Local Zones, AWS Outposts, and Wavelength Zones**

Multiple physical locations for your resources, such as instances and Amazon EBS volumes.

**Security groups**

A virtual firewall that allows you to specify the protocols, ports, and source IP ranges that can reach your instances, and the destination IP ranges to which your instances can connect.

**Elastic IP addresses**

Static IPv4 addresses for dynamic cloud computing.

**Tags**

Metadata that you can create and assign to your Amazon EC2 resources.

**Virtual private clouds (VPCs)**

Virtual networks you can create that are logically isolated from the rest of the AWS Cloud. You can optionally connect these virtual networks to your own network.

# Instance types

When you launch an instance, the instance type that you specify determines the hardware of the host computer used for your instance. Each instance type offers different compute, memory, and storage capabilities, and is grouped in an instance family based on these capabilities. Select an instance type based on the requirements of the application or software that you plan to run on your instance.

Amazon EC2 dedicates some resources of the host computer, such as CPU, memory, and instance storage, to a particular instance. Amazon EC2 shares other resources of the host computer, such as the network and the disk subsystem, among instances. If each instance on a host computer tries to use as much of one of these shared resources as possible, each receives an equal share of that resource. However, when a resource is underused, an instance can consume a higher share of that resource while it's available.

Each instance type provides higher or lower minimum performance from a shared resource. For example, instance types with high I/O performance have a larger allocation of shared resources. Allocating a larger share of shared resources also reduces the variance of I/O performance. For most applications, moderate I/O performance is more than enough. However, for applications that require greater or more consistent I/O performance, consider an instance type with higher I/O performance.

**General purpose Instances:**

**General purpose instances** provide a balance of compute, memory, and networking resources. You can use them for a variety of workloads, such as:

* application servers
* gaming servers
* backend servers for enterprise applications
* small and medium databases

**Compute optimizeed Instances :**

**Compute optimized instances** are ideal for compute-bound applications that benefit from high-performance processors. Like general purpose instances, you can use compute optimized instances for workloads such as web, application, and gaming servers.

However, the difference is compute optimized applications are ideal for high-performance web servers, compute-intensive applications servers, and dedicated gaming servers. You can also use compute optimized instances for batch processing workloads that require processing many transactions in a single group.

**Memory optimized Instances:**

**Memory optimized instances** are designed to deliver fast performance for workloads that process large datasets in memory. In computing, memory is a temporary storage area. It holds all the data and instructions that a central processing unit (CPU) needs to be able to complete actions. Before a computer program or application is able to run, it is loaded from storage into memory. This preloading process gives the CPU direct access to the computer program.

Suppose that you have a workload that requires large amounts of data to be preloaded before running an application. This scenario might be a high-performance database or a workload that involves performing real-time processing of a large amount of unstructured data. In these types of use cases, consider using a memory optimized instance. Memory optimized instances enable you to run workloads with high memory needs and receive great performance.

**Aceelerated computing Instances:**

**Accelerated computing instances** use hardware accelerators, or coprocessors, to perform some functions more efficiently than is possible in software running on CPUs. Examples of these functions include floating-point number calculations, graphics processing, and data pattern matching.

In computing, a hardware accelerator is a component that can expedite data processing. Accelerated computing instances are ideal for workloads such as graphics applications, game streaming, and application streaming.

**Storage optimized Instances:**

**Storage optimized instances** are designed for workloads that require high, sequential read and write access to large datasets on local storage. Examples of workloads suitable for storage optimized instances include distributed file systems, data warehousing applications, and high-frequency online transaction processing (OLTP) systems.

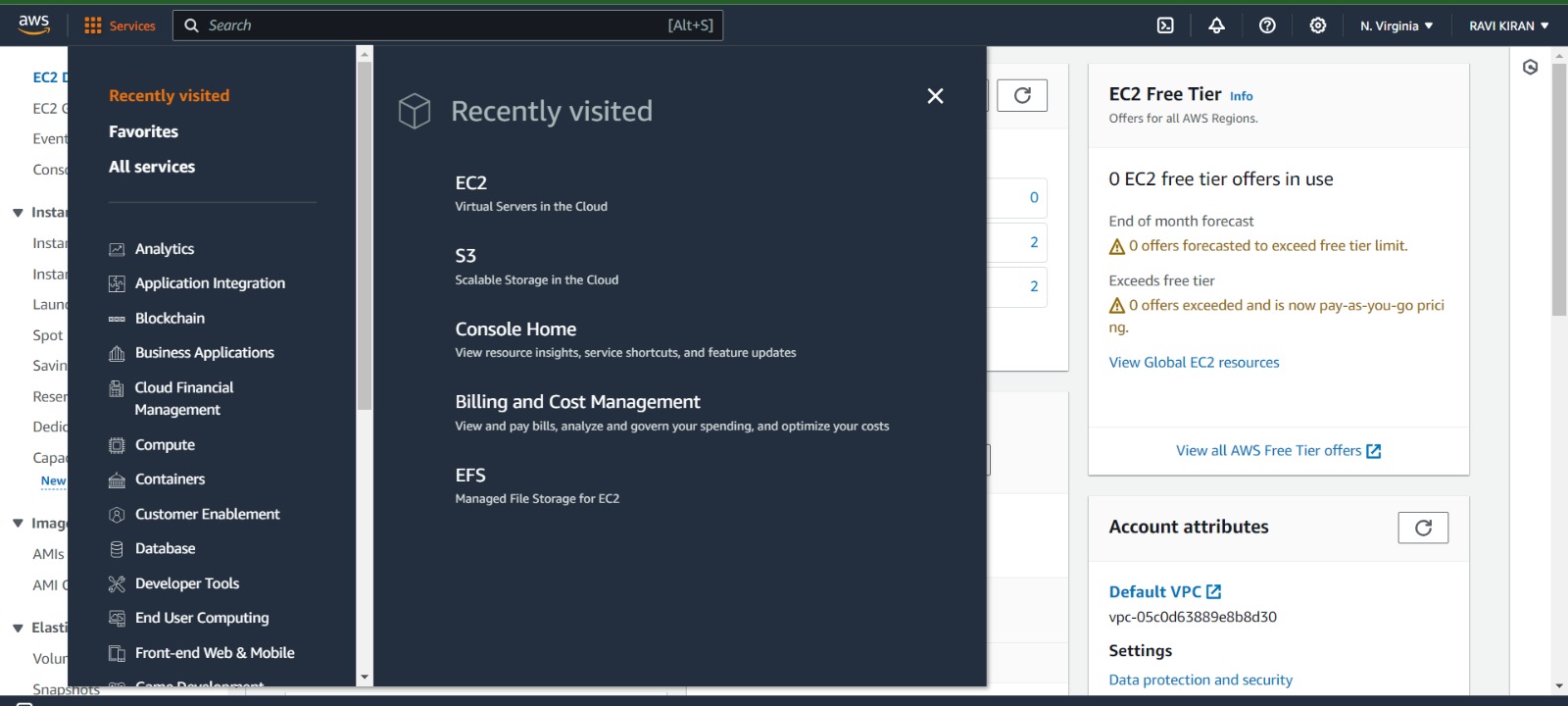
In computing, the term input/output operations per second (IOPS) is a metric that measures the performance of a storage device. It indicates how many different input or output operations a device can perform in one second. Storage optimized instances are designed to deliver tens of thousands of low-latency, random IOPS to applications.

You can think of input operations as data put into a system, such as records entered into a database. An output operation is data generated by a server. An example of output might be the analytics performed on the records in a database. If you have an application that has a high IOPS requirement, a storage optimized instance can provide better performance over other instance types not optimized for this kind of use case.

**CREATING AN EC2 INSTANCE.**

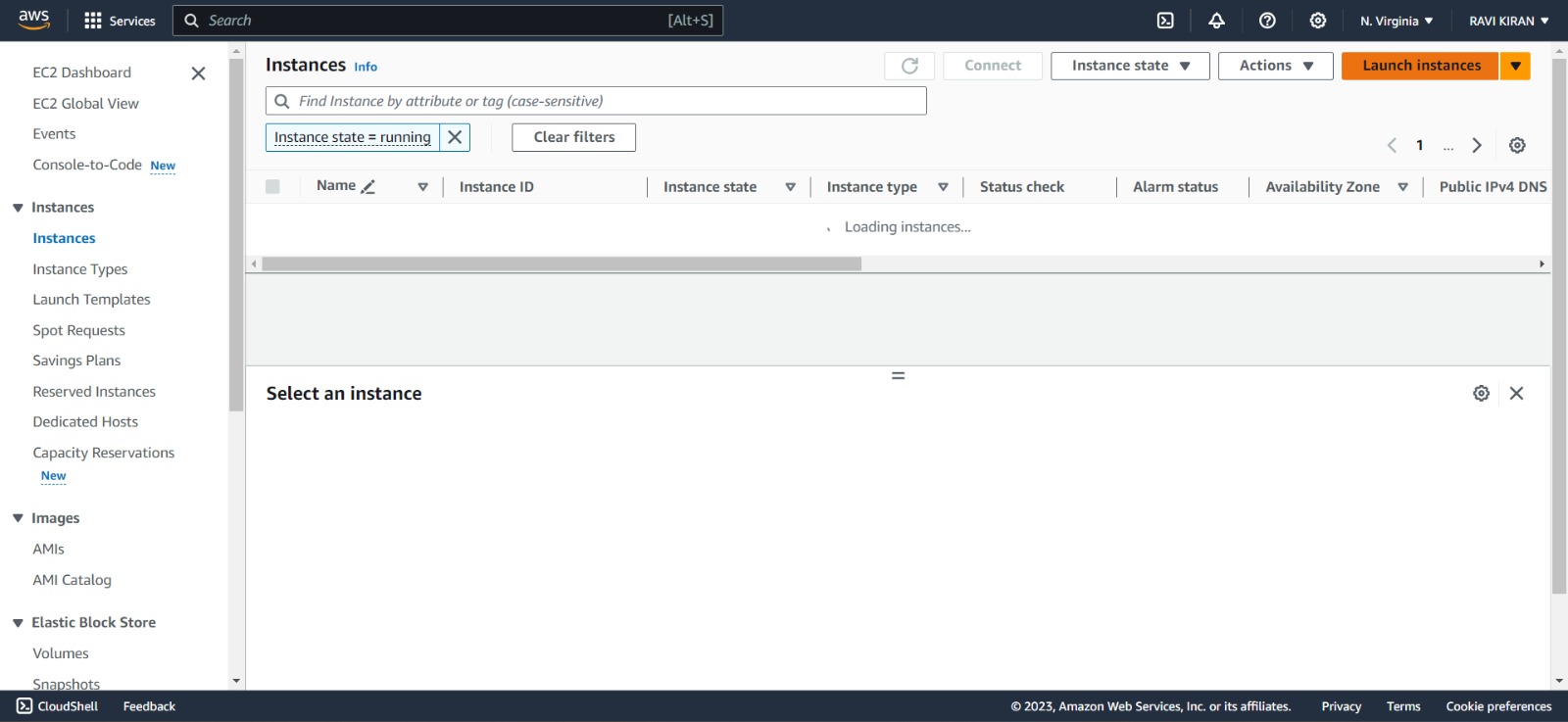
**Step-1:**

Open the Amazon EC2 console at AWS



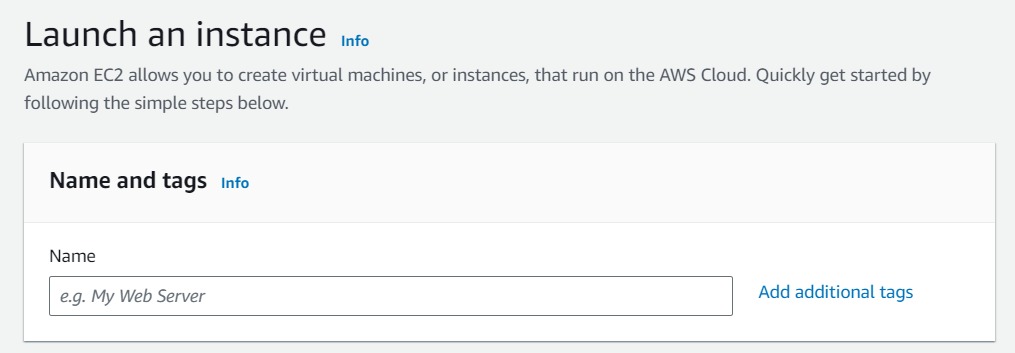
**Step-2:**

From the EC2 console dashboard, in the Launch instance box, choose Launch instance, and then choose Launch instance from the options that appear.



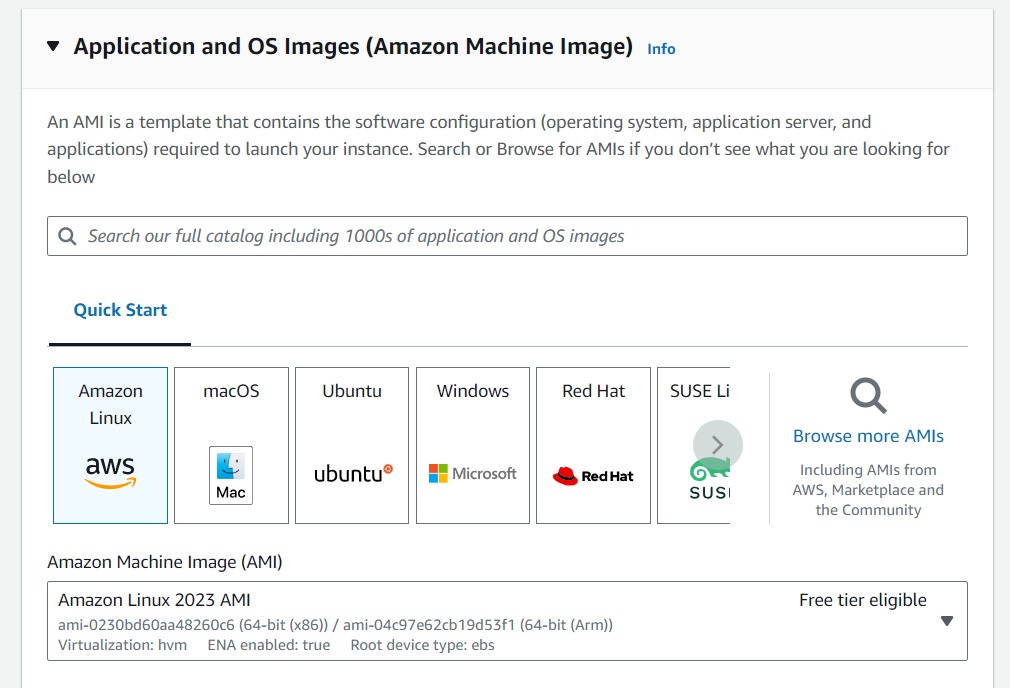
**Step-3:**

Under Name and tags, for Name, enter a descriptive name for your instance**.**



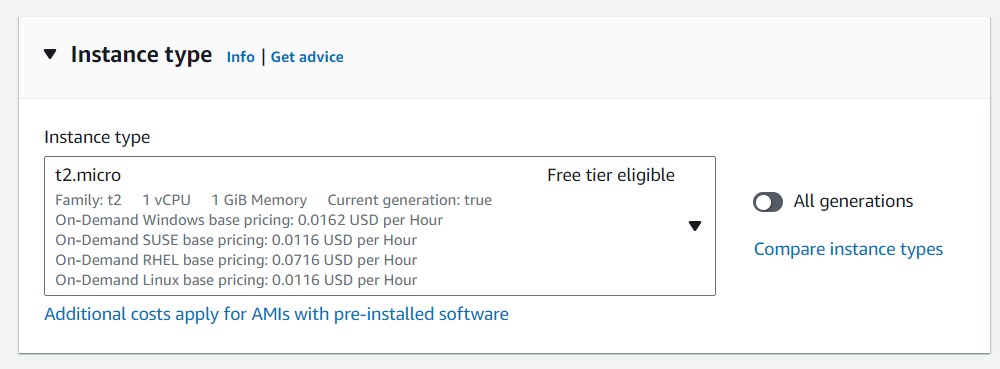
**Step-4:**

Under Application and OS Images (Amazon Machine Image), choose the operating system image and version (Amazon Machine Image-AMI) of the Linux distribution you wish to use.

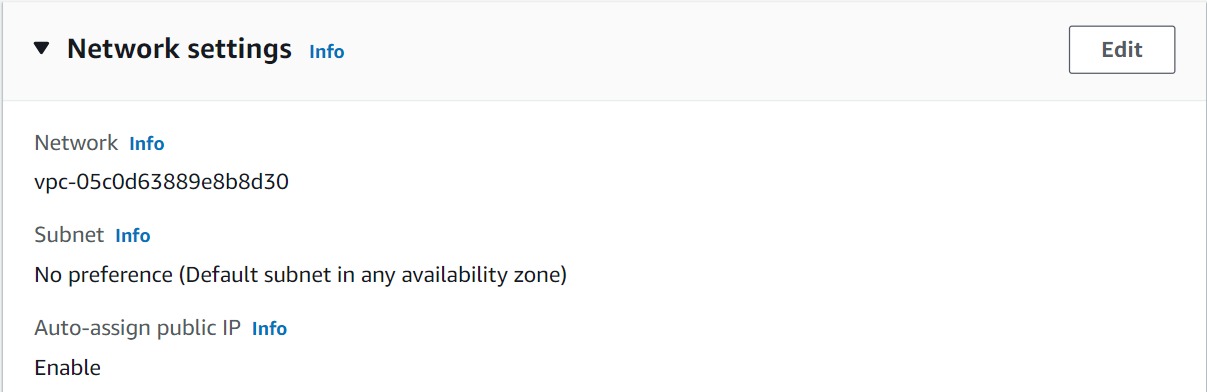


**Step-5:**

Choose an instance type. The instance type determines the hardware of the host computer used for your instance.

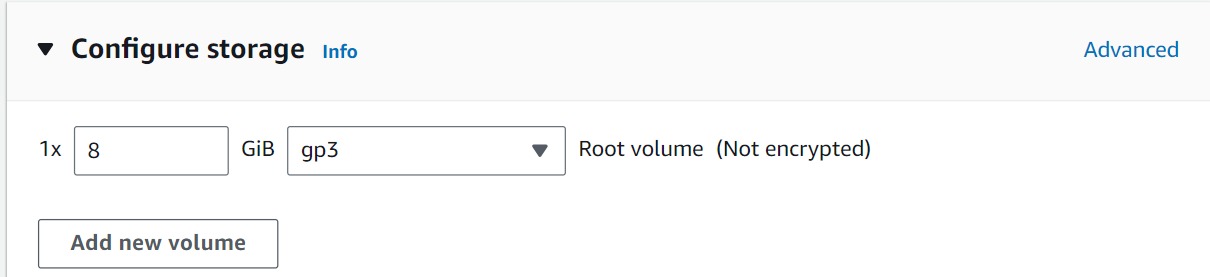


**Step-6:**

Configure the instance details. You can configure network settings, add storage, and specify security groups. 

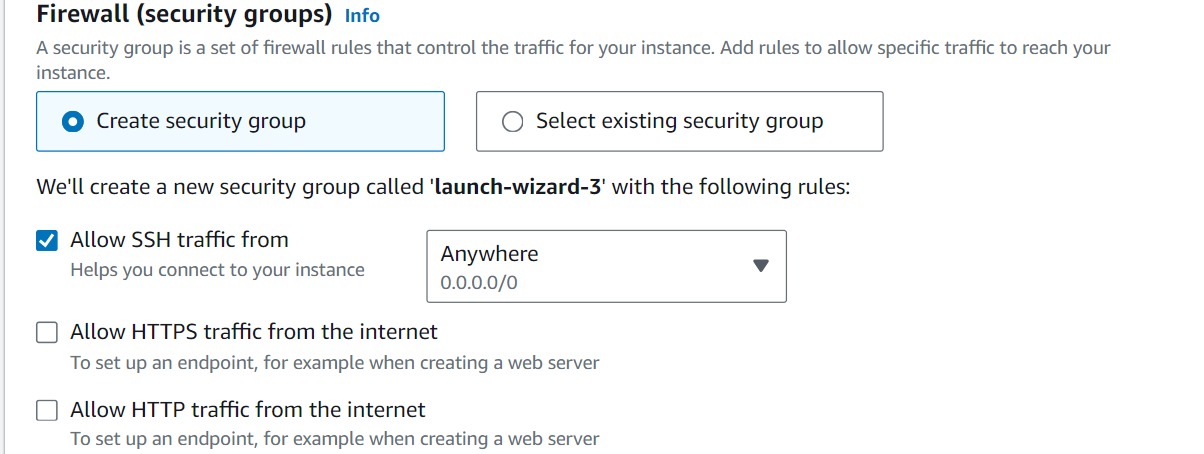
**Step-7:**

Add storage. You can add storage to your instance by adding Amazon Elastic Block Store (Amazon EBS) volumes.



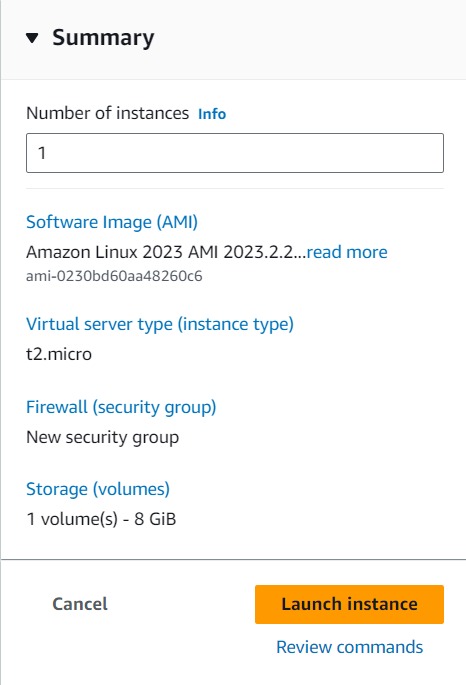
**Step-8:**

Configure security group. A security group acts as a virtual firewall for your instance to control inbound and outbound traffic.

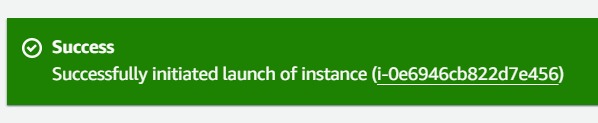


**Step-9:**

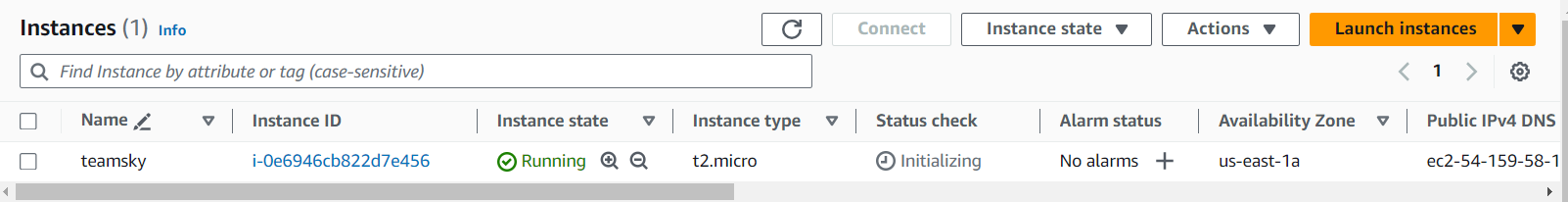
Review and launch. Review the instance details and click on the Launch button.



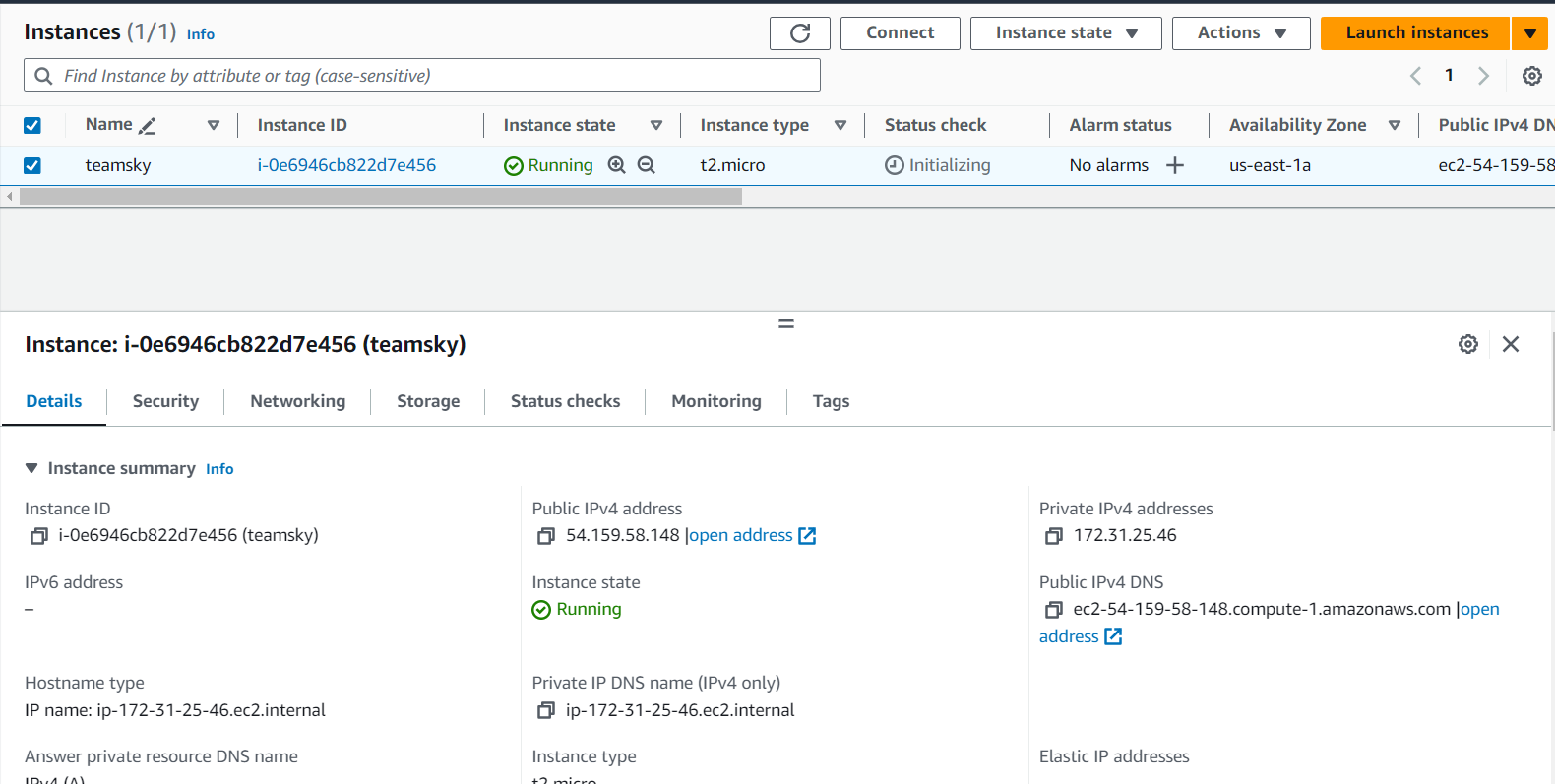
The pop-up appears once the Instance is created successfully.



**Step-10:**

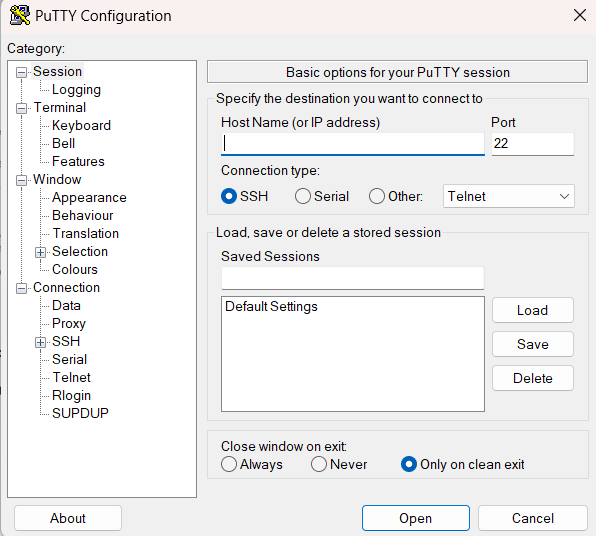
 Find your created instance in the instances and make sure that it is in running status**.**

**Step-11:**

Select the created instance to see its properties. 

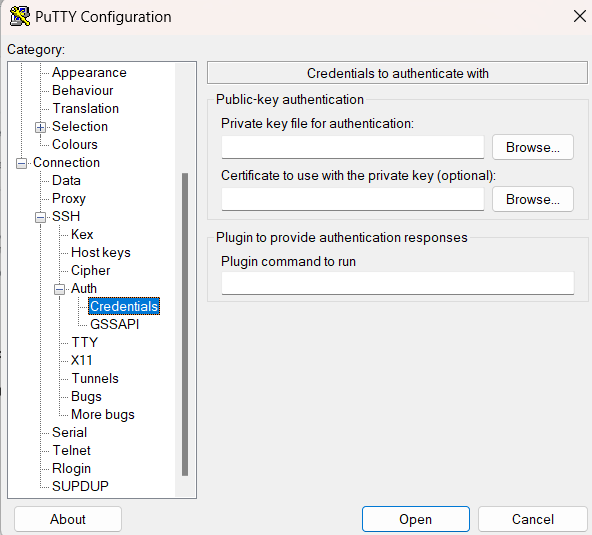
**Step-12:**

Upload the Public IPv4 address in PuTTY Gen



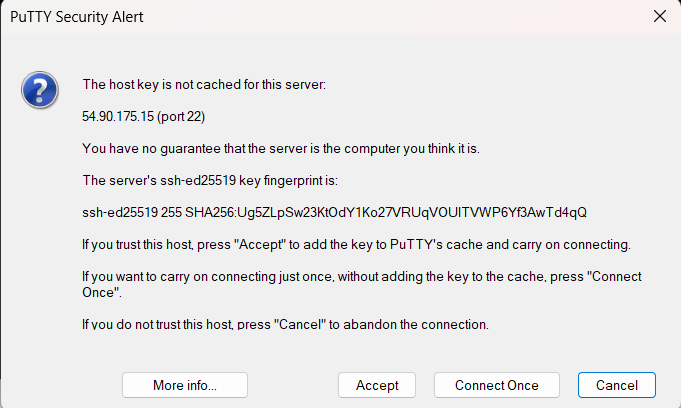
**Step-13:**

Now under Connections-SSH-Auth, select Credentials. There browse the .ppk key pair downloaded file.

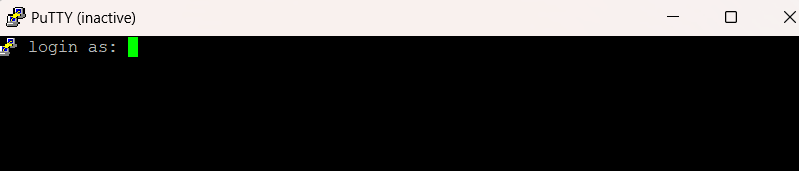


**Step-14:**

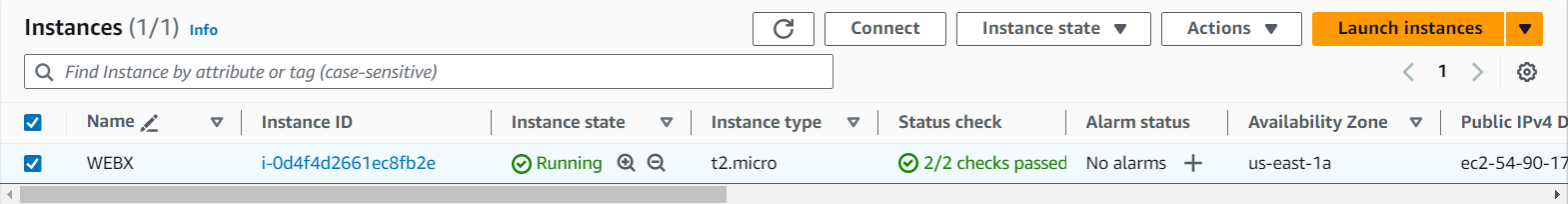
Then click launch then a pop-up appears, click “Accept”.



**Step-15:**

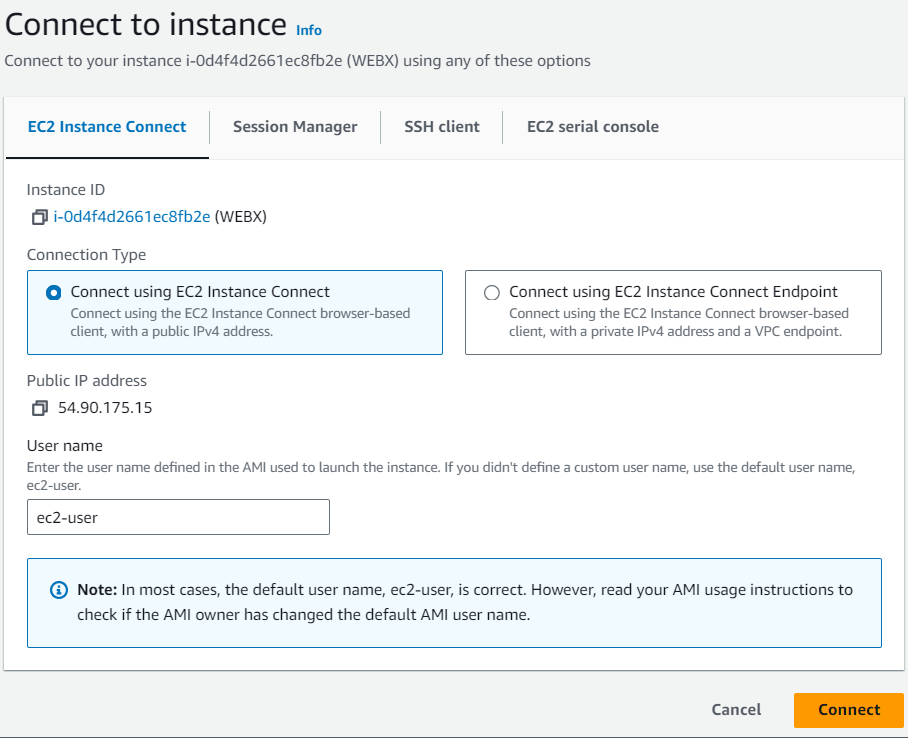
Then the command prompt appears.

**Step-16:**

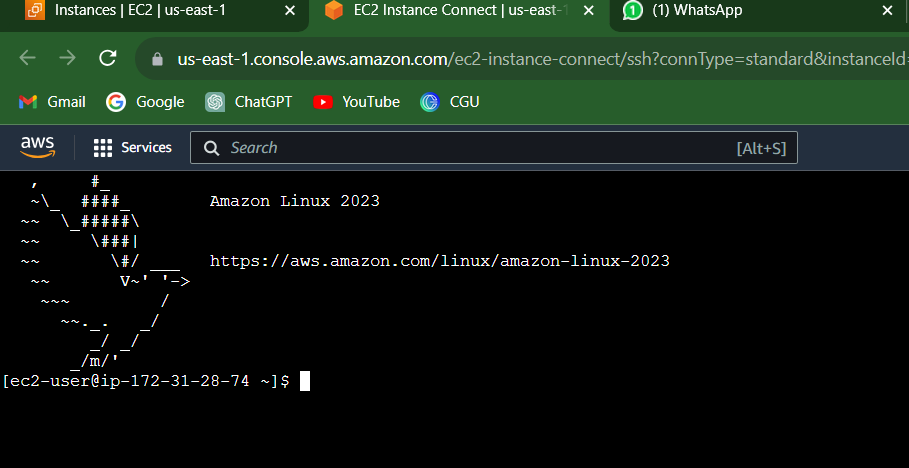
Click “Connect” for Connecting without the use of PuTTY.

**Step-17:**

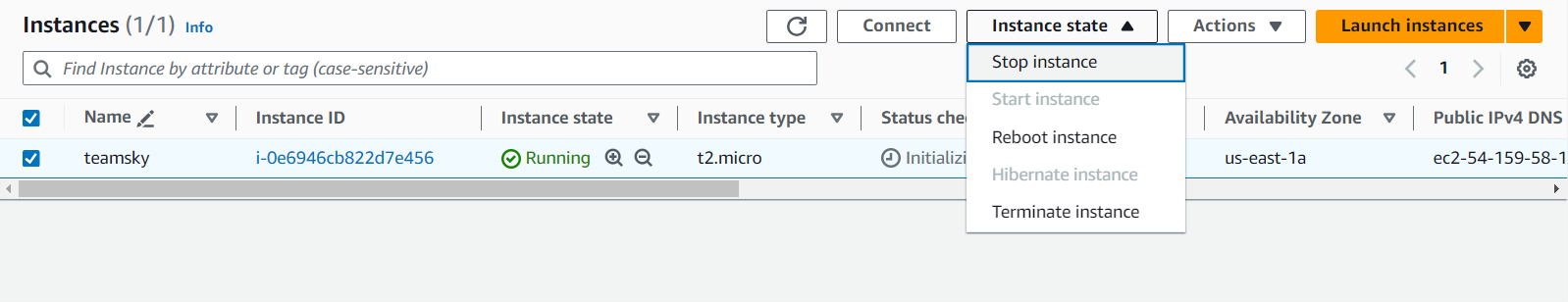
Select the required connection type, enter the username and the click “Connect”.



**Step-18:**

After clicking “Connect”. It will redirect to a new window, there we can give the required commands.

**Step-19:**

Go to Instance State to delete/stop/terminate the created Instance.