Task 3. Build a VPC with Public & Private Subnets

Goal: Deploy infrastructure using networking best practices. Tasks:

· Create a VPC with 2 public and 2 private subnets.

· Configure route tables and Internet Gateway for public subnets.

· Set up a NAT Gateway for private subnets.

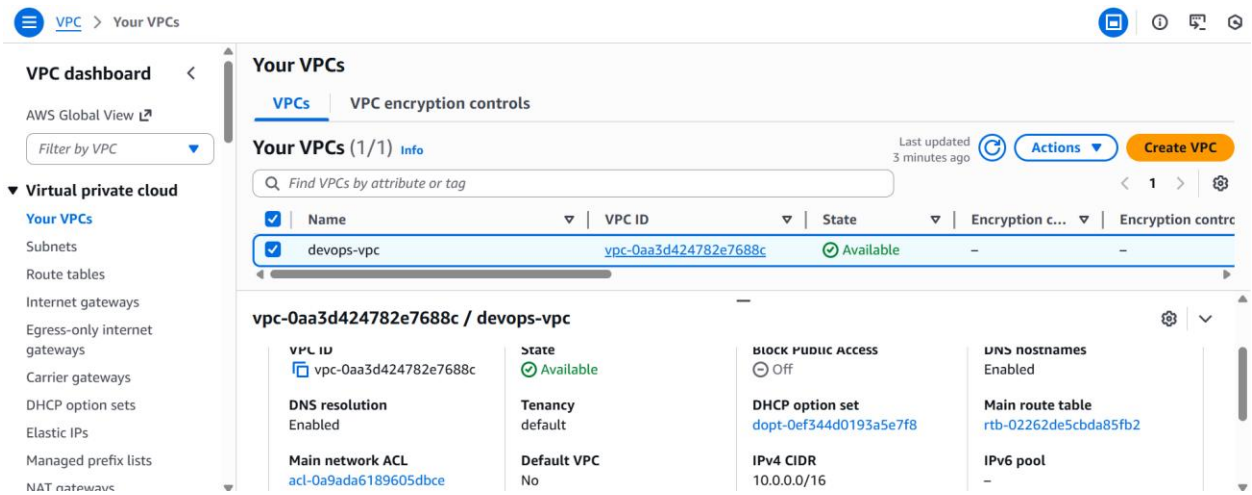
· Launch a public Bastion Host (EC2).

· Launch a private EC2 instance and connect to it through the Bastion using SSH.

➔

1. Create vpc

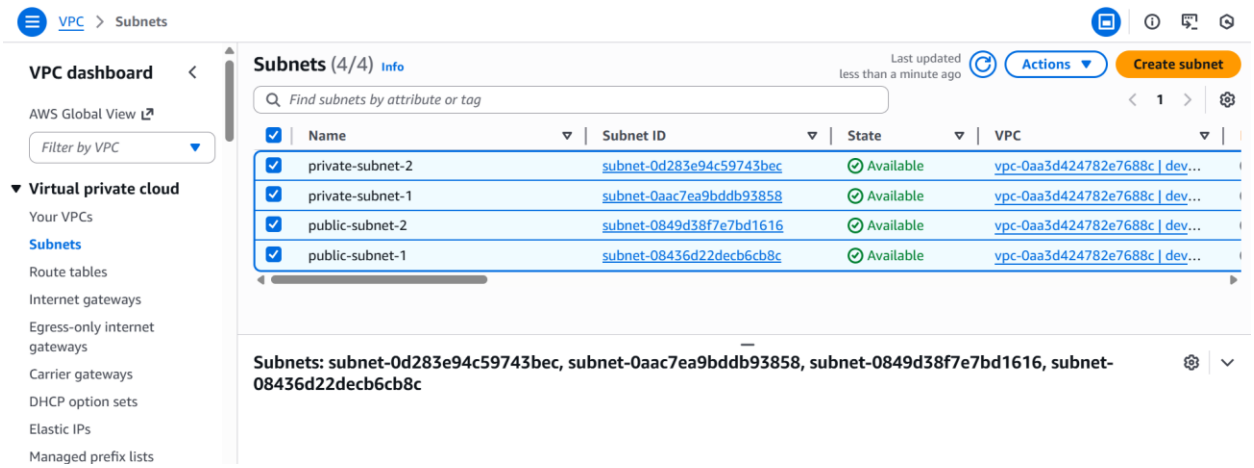
```
PS C:\Users\112256\k8\task\task-3\vpc-privatepublic-sub> aws ec2 describe-vpcs --filters Name=tag:Name,Values=devops-vpc
{
    "Vpcs": [
        {
            "OwnerId": "442955307475",
            "InstanceTenancy": "default",
            "CidrBlockAssociationSet": [
                {
                    "AssociationId": "vpc-cidr-assoc-00643a86f82737128",
                    "CidrBlock": "10.0.0.0/16",
                    "CidrBlockState": {
                        "State": "associated"
                    }
                }
            ],
            "IsDefault": false,
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "devops-vpc"
                }
            ],
            "BlockPublicAccessStates": {
                "InternetGatewayBlockMode": "off"
            },
            "VpcId": "vpc-0aa3d424782e7688c",
            "State": "available",
            "CidrBlock": "10.0.0.0/16",
```



2. Create public and private subnet
-Public subnets → internet-facing resources
- Private subnets → internal workloads

```
PS C:\Users\112256\k8\task\task-3\vpc-privatepublic-sub> aws ec2 describe-subnets --filters Name=vpc-id,Values=vpc-0aa3d42
4782e7688c
{
    "Subnets": [
        {
            "AvailabilityZoneId": "use1-az2",
            "MapCustomerOwnedIpOnLaunch": false,
            "OwnerId": "442955307475",
            "AssignIpv6AddressOnCreation": false,
            "Ipv6CidrBlockAssociationSet": [],
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "private-subnet-2"
                }
            ],
            "SubnetArn": "arn:aws:ec2:us-east-1:442955307475:subnet/subnet-0d283e94c59743bec",
            "EnableDns64": false,
            "Ipv6Native": false,
            "PrivateDnsNameOptionsOnLaunch": {
                "HostnameType": "ip-name",
                "EnableResourceNameDnsARecord": false,
                "EnableResourceNameDnsAAAARecord": false
            },
            "BlockPublicAccessStates": {
                "InternetGatewayBlockMode": "off"
            },
```

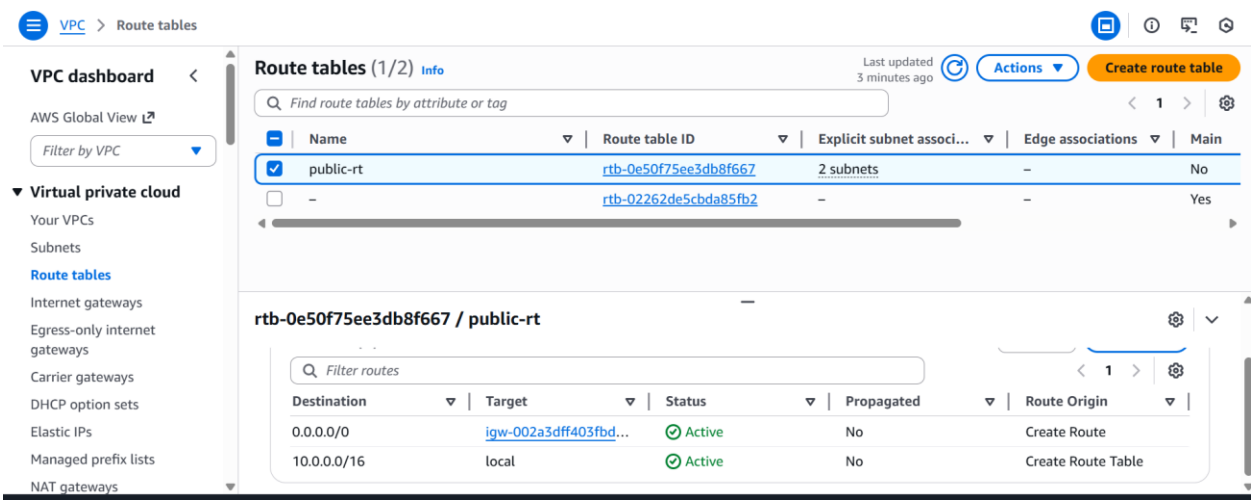
```
    },
    "SubnetId": "subnet-0d283e94c59743bec",
    "State": "available",
    "VpcId": "vpc-0aa3d424782e7688c",
    "CidrBlock": "10.0.12.0/24",
    "AvailableIpAddressCount": 251,
    "AvailabilityZone": "us-east-1b",
    "DefaultForAz": false,
    "MapPublicIpOnLaunch": false
},
{
    "AvailabilityZoneId": "use1-az1",
    "MapCustomerOwnedIpOnLaunch": false,
    "OwnerId": "442955307475",
    "AssignIpv6AddressOnCreation": false,
    "Ipv6CidrBlockAssociationSet": [],
    "Tags": [
        {
            "Key": "Name",
            "Value": "private-subnet-1"
        }
    ],
    "SubnetArn": "arn:aws:ec2:us-east-1:442955307475:subnet/subnet-0aac7ea9bddb93858",
    "EnableDns64": false,
    "Ipv6Native": false,
    "PrivateDnsNameOptionsOnLaunch": {
        "HostnameType": "ip-name",
    "MapCustomerOwnedIpOnLaunch": false,
    "OwnerId": "442955307475",
    "AssignIpv6AddressOnCreation": false,
    "Ipv6CidrBlockAssociationSet": [],
    "Tags": [
        {
            "Key": "Name",
            "Value": "public-subnet-2"
        }
    ],
    "SubnetArn": "arn:aws:ec2:us-east-1:442955307475:subnet/subnet-0849d38f7e7bd1616",
    "EnableDns64": false,
    "Ipv6Native": false,
    "PrivateDnsNameOptionsOnLaunch": {
        "HostnameType": "ip-name",
        "EnableResourceNameDnsARecord": false,
        "EnableResourceNameDnsAAAARecord": false
    },
    "BlockPublicAccessStates": {
        "InternetGatewayBlockMode": "off"
    },
    "SubnetId": "subnet-0849d38f7e7bd1616",
    "State": "available",
    "VpcId": "vpc-0aa3d424782e7688c",
    "CidrBlock": "10.0.2.0/24",
    "AvailableIpAddressCount": 251,
    "AvailabilityZone": "us-east-1b",
```

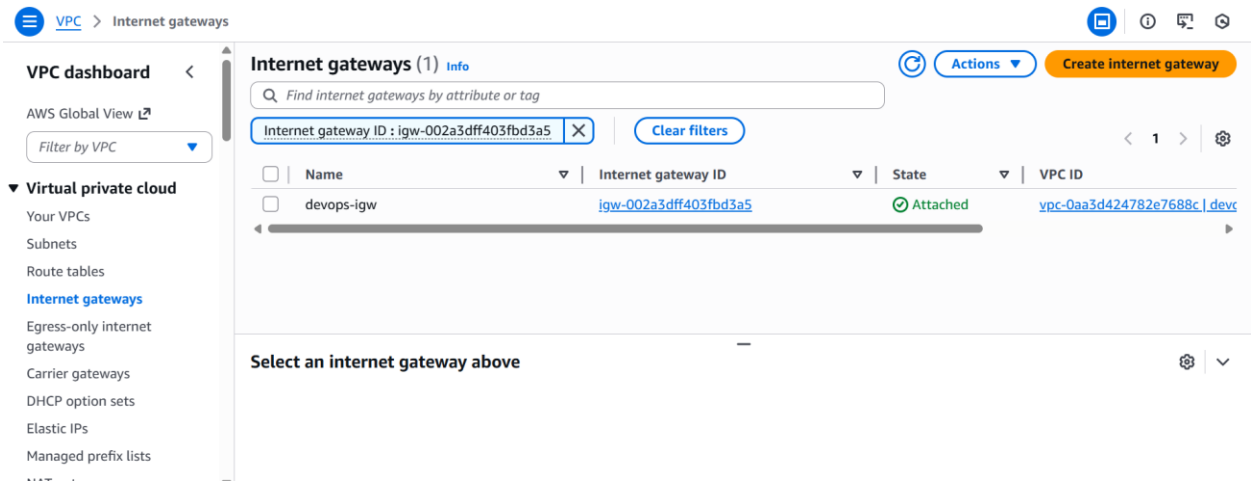


3.Configure route tables and Internet Gateway for public subnets.

```
PS C:\Users\112256\k8\task\task-3\vpc-privatepublic-sub> aws ec2 describe-route-tables --filters Name=tag:Name,Values=publ
ic-rt
{
    "RouteTables": [
        {
            "Associations": [
                {
                    "Main": false,
                    "RouteTableAssociationId": "rtbassoc-00a97f2b60afe9a0a",
                    "RouteTableId": "rtb-0e50f75ee3db8f667",
                    "SubnetId": "subnet-0849d38f7e7bd1616",
                    "AssociationState": {
                        "State": "associated"
                    }
                },
                {
                    "Main": false,
                    "RouteTableAssociationId": "rtbassoc-0f3a2c4089de37fef",
                    "RouteTableId": "rtb-0e50f75ee3db8f667",
                    "SubnetId": "subnet-08436d22decb6cb8c",
                    "AssociationState": {
                        "State": "associated"
                    }
```

```
],
"PropagatingVgws": [],
"RouteTableId": "rtb-0e50f75ee3db8f667",
"Routes": [
    {
        "DestinationCidrBlock": "10.0.0.0/16",
        "GatewayId": "local",
        "Origin": "CreateRouteTable",
        "State": "active"
    },
    {
        "DestinationCidrBlock": "0.0.0.0/0",
        "GatewayId": "igw-002a3dff403fbd3a5",
        "Origin": "CreateRoute",
        "State": "active"
    }
],
"Tags": [
    {
        "Key": "Name",
        "Value": "public-rt"
    }
],
```

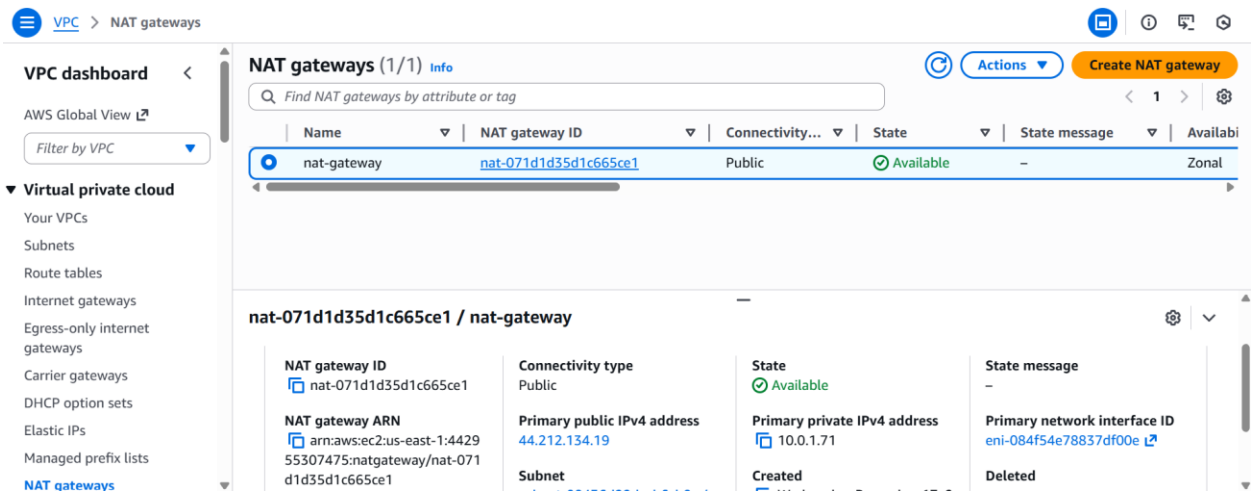





4. Set up a NAT Gateway for private subnets.

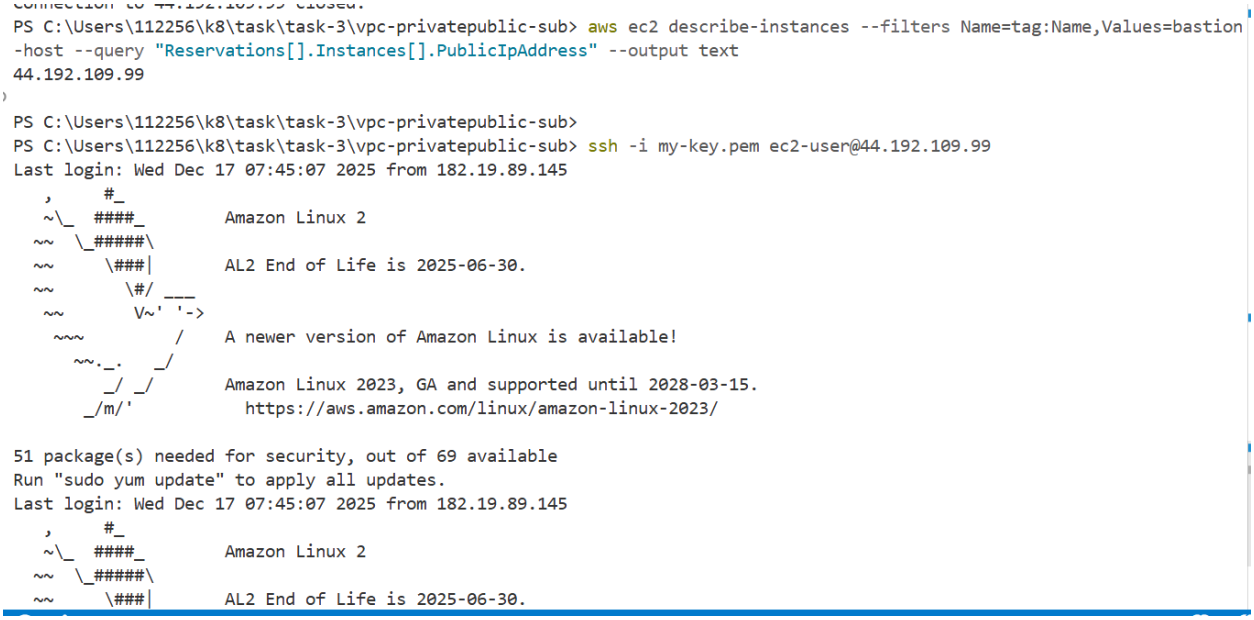
Use - Private instances need outbound internet access securely.

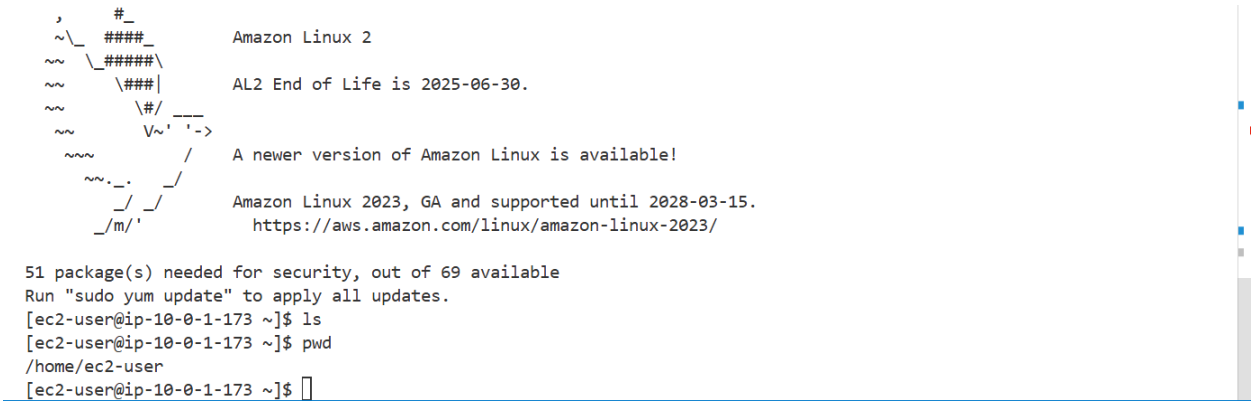
```
PS C:\Users\112256\k8\task\task-3\vpc-privatepublic-sub> aws ec2 describe-nat-gateways
{
    "NatGateways": [
        {
            "CreateTime": "2025-12-17T06:56:03+00:00",
            "NatGatewayAddresses": [
                {
                    "AllocationId": "eipalloc-066f396ae87b4587c",
                    "NetworkInterfaceId": "eni-084f54e78837df00e",
                    "PrivateIp": "10.0.1.71",
                    "PublicIp": "44.212.134.19",
                    "AssociationId": "eipassoc-017930b80b955982e",
                    "IsPrimary": true,
                    "Status": "succeeded"
                }
            ],
            "NatGatewayId": "nat-071d1d35d1c665ce1",
            "State": "available",
            "SubnetId": "subnet-08436d22decb6cb8c",
            "VpcId": "vpc-0aa3d424782e7688c",
            "Tags": [
                {
                    "Key": "Name",
                    "Value": "nat-gateway"
                }
            ],
            "ConnectivityType": "public"
        }
    ]
}
```

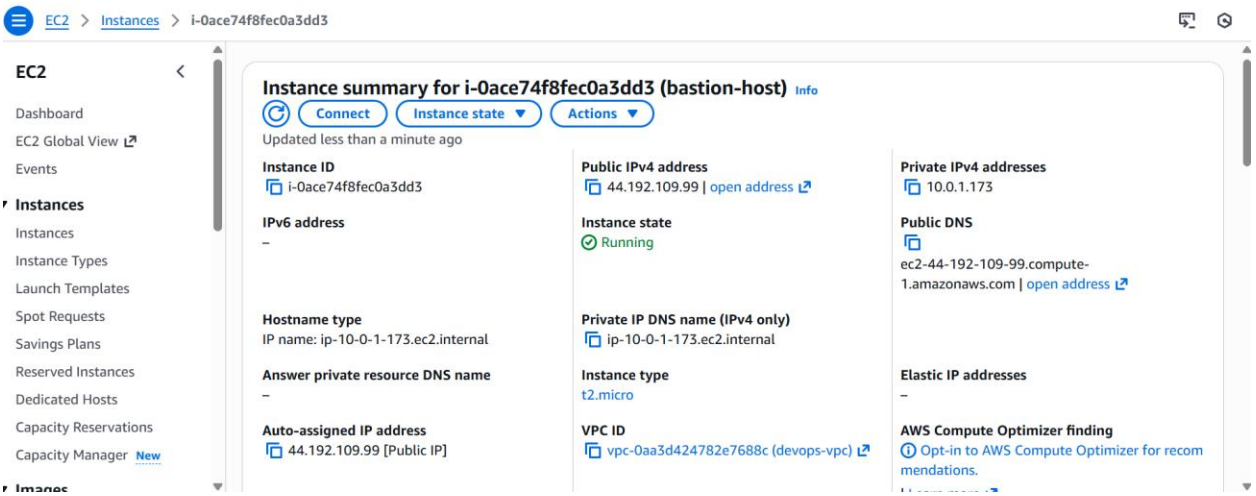


5.Launch a public Bastion Host (EC2).

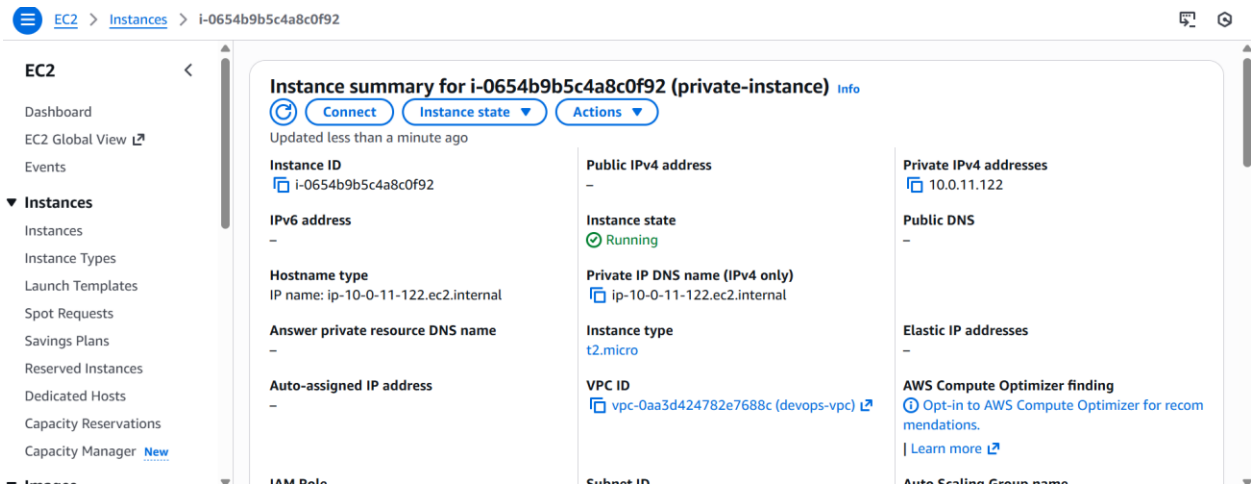
- Use for - Single secure SSH entry point.







6.Launch a private EC2 instance and connect to it through the Bastion using SSH.



```
PS C:\Users\112256\k8\task\task-3\vpc-privatepublic-sub> ssh -i .\my-key.pem ec2-user@44.192.109.99
Last login: Wed Dec 17 07:50:01 2025 from 182.19.89.145
   ,     #_
   ~\_  ####_        Amazon Linux 2
  ~~  \_#####\
  ~~     \###|       AL2 End of Life is 2025-06-30.
  ~~       \#/ ___
   ~~       V~' '->
    ~~~         /    A newer version of Amazon Linux is available!
      ~~._.   _/
         _/ _/       Amazon Linux 2023, GA and supported until 2028-03-15.
       _/m/'           https://aws.amazon.com/linux/amazon-linux-2023/

51 package(s) needed for security, out of 69 available
Run "sudo yum update" to apply all updates.
[ec2-user@ip-10-0-1-173 ~]$ ls
[ec2-user@ip-10-0-1-173 ~]$ ssh ec2-user@10.0.11.122
The authenticity of host '10.0.11.122 (10.0.11.122)' can't be established.
ECDSA key fingerprint is SHA256:m5dquWQWaXfWLJdJUfD/gNoX90t2ctHbDycaTQ44ktY.
ECDSA key fingerprint is MD5:c1:e3:93:2a:df:f8:ed:05:61:d6:ee:17:85:bc:50:9d.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '10.0.11.122' (ECDSA) to the list of known hosts.
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-0-1-173 ~]$
```

```
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
[ec2-user@ip-10-0-1-173 ~]$ ping google.com
PING google.com (142.251.163.101) 56(84) bytes of data.
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=1 ttl=102 time=2.04 ms
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=2 ttl=102 time=2.05 ms
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=3 ttl=102 time=2.01 ms
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=4 ttl=102 time=2.05 ms
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=5 ttl=102 time=2.04 ms
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=6 ttl=102 time=2.03 ms
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=7 ttl=102 time=2.03 ms
64 bytes from wv-in-f101.1e100.net (142.251.163.101): icmp_seq=8 ttl=102 time=2.08 ms
```