

launch Ec2 instance with the AWS CLI IN public subnet and access it over the internet ?

---

## 1. Create a VPC

**aws ec2 create-vpc --cidr-block 10.0.0.0/16**

- This command creates a **Virtual Private Cloud (VPC)** with a CIDR

```
PS C:\WINDOWS\system32> aws ec2 create-vpc --cidr-block 10.0.0.0/16
{
  "Vpc": {
    "OwnerId": "442042536659",
    "InstanceTenancy": "default",
    "Ipv6CidrBlockAssociationSet": [],
    "CidrBlockAssociationSet": [
      {
        "AssociationId": "vpc-cidr-assoc-0b5902b9b5b4020e4",
        "CidrBlock": "10.0.0.0/16",
        "CidrBlockState": {
          "State": "associated"
        }
      }
    ],
    "IsDefault": false,
    "VpcId": "vpc-062e6bb51aa3d8fd0",
    "State": "pending",
    "CidrBlock": "10.0.0.0/16",
    "DhcpOptionsId": "dopt-05c79df72539e51cc"
  }
}
```

## 2. Create a Subnet

**aws ec2 create-subnet --vpc-id vpc-062e6bb51aa3d8fd0 --cidr-block 10.0.1.0/24**

- This command creates a **subnet** within the VPC (vpc-062e6bb51aa3d8fd0) with a CIDR block of 10.0.1.0/24 (256 IP addresses: 10.0.1.0 to 10.0.1.255).

```
PS C:\WINDOWS\system32> aws ec2 create-subnet --vpc-id vpc-062e6bb51aa3d8fd0 --cidr-block 10.0.1.0/24
{
  "Subnet": {
    "AvailabilityZoneId": "aps1-az2",
    "OwnerId": "442042536659",
    "AssignIpv6AddressOnCreation": false,
    "Ipv6CidrBlockAssociationSet": [],
    "SubnetArn": "arn:aws:ec2:ap-south-1:442042536659:subnet/subnet-0a718e5c7fae5d233",
    "EnableDns64": false,
    "Ipv6Native": false,
    "PrivateDnsNameOptionsOnLaunch": {
      "HostnameType": "ip-name",
      "EnableResourceNameDnsARecord": false,
      "EnableResourceNameDnsAAAARecord": false
    },
    "SubnetId": "subnet-0a718e5c7fae5d233",
    "State": "available",
    "VpcId": "vpc-062e6bb51aa3d8fd0",
    "CidrBlock": "10.0.1.0/24",
    "AvailableIpAddressCount": 251,
    "AvailabilityZone": "ap-south-1c",
    "DefaultForAz": false,
    "MapPublicIpOnLaunch": false
  }
}
```

## 3. Create an Internet Gateway

**aws ec2 create-internet-gateway**

- This command creates an **Internet Gateway (IGW)** to allow instances in the VPC to access the internet

```
PS C:\WINDOWS\system32> aws ec2 create-internet-gateway
{
  "InternetGateway": {
    "Attachments": [],
    "InternetGatewayId": "igw-0238a73acb4c05ad8",
    "OwnerId": "442042536659",
    "Tags": []
  }
}
```

---

#### 4. Attach the Internet Gateway to the VPC

**aws ec2 attach-internet-gateway --internet-gateway-id igw-0238a73acb4c05ad8 --vpc-id vpc-062e6bb51aa3d8fd0**

- This command attaches the **Internet Gateway** (igw-0238a73acb4c05ad8) to the **VPC** (vpc-062e6bb51aa3d8fd0).

```
PS C:\WINDOWS\system32> aws ec2 attach-internet-gateway --internet-gateway-id igw-0238a73acb4c05ad8 --vpc-id vpc-062e6bb51aa3d8fd0
```

---

#### 5. Create a Route Table

**aws ec2 create-route-table --vpc-id vpc-062e6bb51aa3d8fd0**

- This command creates a **Route Table** for the **VPC** (vpc-062e6bb51aa3d8fd0).

```
PS C:\WINDOWS\system32> aws ec2 create-route-table --vpc-id vpc-062e6bb51aa3d8fd0
{
  "RouteTable": {
    "Associations": [],
    "PropagatingVgws": [],
    "RouteTableId": "rtb-0b26f600624bf8a89",
    "Routes": [
      {
        "DestinationCidrBlock": "10.0.0.0/16",
        "GatewayId": "local",
        "Origin": "CreateRouteTable",
        "State": "active"
      }
    ],
    "Tags": [],
    "VpcId": "vpc-062e6bb51aa3d8fd0",
    "OwnerId": "442042536659"
  },
  "ClientToken": "fa14ae4d-2668-4103-b94e-caa26f4ddbea"
}
```

---

#### 6. Add a Route to the Internet Gateway

**aws ec2 create-route --route-table-id rtb-0b26f600624bf8a89 --destination-cidr-block 0.0.0.0/0 --gateway-id igw-0238a73acb4c05ad8**

- This command adds a route to the **Route Table** (rtb-0b26f600624bf8a89).

```
PS C:\WINDOWS\system32> aws ec2 create-route --route-table-id rtb-0b26f600624bf8a89 --destination-cidr-block 0.0.0.0/0 --gateway-id igw-0238a73acb4c05ad8
{
  "Return": true
}
```

---

#### 7. Associate the Route Table with the Subnet

**aws ec2 associate-route-table --route-table-id rtb-0b26f600624bf8a89 --subnet-id subnet-0a718e5c7fae5d233**

- This command **associates the Route Table** (rtb-0b26f600624bf8a89) with the **Subnet** (subnet-0a718e5c7fae5d233).

```
PS C:\WINDOWS\system32> aws ec2 associate-route-table --route-table-id rtb-0b26f600624bf8a89 --subnet-id subnet-0a718e5c7fae5d233
{
  "AssociationId": "rtbassoc-0343c9ba5d72a345a",
  "AssociationState": {
    "State": "associated"
  }
}
```

## 8. Launch an EC2 Instance

```
aws ec2 run-instances --image-id ami-00bb6a80f01f03502 --instance-type t2.micro
--key-name Ujwal-CLI --security-group-ids sg-0d01d83ca14f2049a --subnet-id
subnet-0a718e5c7fae5d233 --associate-public-ip-address
```

```
PS C:\WINDOWS\system32> aws ec2 run-instances --image-id ami-00bb6a80f01f03502 --instance-type t2.micro --key-name Ujwal-CLI --security-group-ids sg-0d01d83ca14f2049a -
associate-public-ip-address
{
  "ReservationId": "r-05d6010d95098eeb5",
  "OwnerId": "442042536659",
  "Groups": [],
  "Instances": [
    {
      "Architecture": "x86_64",
      "BlockDeviceMappings": [],
      "ClientToken": "6d3869d5-e9ea-4b2a-9db7-67ef83fa20fd",
      "EbsOptimized": false,
      "EnaSupport": true,
      "Hypervisor": "Xen",
      "NetworkInterfaces": [
        {
          "Attachment": {
            "AttachTime": "2025-02-28T14:18:00+00:00",
            "AttachmentId": "eni-attach-0dace18c19411a286",
            "DeleteOnTermination": true,
            "DeviceIndex": 0,
            "Status": "attaching",
            "NetworkCardIndex": 0
          },
          "Description": "",
          "Groups": [
            {
              "GroupId": "sg-0d01d83ca14f2049a",
              "GroupName": "default"
            }
          ],
          "Ipv6Addresses": [],
          "MacAddress": "0a:ca:a9:1c:9b:01",
          "NetworkInterfaceId": "eni-04f0cfacc40557e58",
          "OwnerId": "442042536659",
          "PrivateDnsName": "ip-172-31-6-33.ap-south-1.compute.internal",
          "PrivateIpAddress": "172.31.6.33",
          "PrivateIpAddresses": [
            {

```

```

    "PrivateDnsName": "ip-172-31-6-33.ap-south-1.compute.internal",
    "PrivateIpAddress": "172.31.6.33",
    "PrivateIpAddresses": [
      {
        "Primary": true,
        "PrivateDnsName": "ip-172-31-6-33.ap-south-1.compute.internal",
        "PrivateIpAddress": "172.31.6.33"
      }
    ],
    "SourceDestCheck": true,
    "Status": "in-use",
    "SubnetId": "subnet-03e8cc288b1bf9e67",
    "VpcId": "vpc-0473ebd347ec8b538",
    "InterfaceType": "interface",
    "Operator": {
      "Managed": false
    }
  },
  "RootDeviceName": "/dev/sda1",
  "RootDeviceType": "ebs",
  "SecurityGroups": [
    {
      "GroupId": "sg-0d81d83ca14f2049a",
      "GroupName": "default"
    }
  ],
  "SourceDestCheck": true,
  "StateReason": {
    "Code": "pending",
    "Message": "pending"
  },
  "VirtualizationType": "hvm",
  "CpuOptions": {
    "CoreCount": 1,
    "ThreadsPerCore": 1
  },
  "CapacityReservationSpecification": {
    "CapacityReservationPreference": "open"
  },
  "MetadataOptions": {
    "State": "pending",
    "HttpTokens": "required",

```

```
Select Administrator: Windows PowerShell

{"BootMode": "uefi-preferred",
 "PrivateDnsNameOptions": {
   "HostNameType": "ip-name",
   "EnableResourceNameDnsARecord": false,
   "EnableResourceNameDnsAAAARecord": false
 },
 "MaintenanceOptions": {
   "AutoRecovery": "default"
 },
 "CurrentInstanceBootMode": "legacy-bios",
 "Operator": {
   "Managed": false
 },
 "InstanceId": "i-08fc042a159d0bd38",
 "ImageId": "ami-00bb0a80f01f03502",
 "State": {
   "Code": 0,
   "Name": "pending"
 },
 "PrivateDnsName": "ip-172-31-6-33.ap-south-1.compute.internal",
 "PublicDnsName": "",
 "StateTransitionReason": "",
 "KeyName": "Ujwal-CLI",
 "AmiLaunchIndex": 0,
 "ProductCodes": [],
 "InstanceType": "t2.micro",
 "LaunchTime": "2025-02-28T14:18:00+00:00",
 "Placement": {
   "GroupName": "",
   "Tenancy": "default",
   "AvailabilityZone": "ap-south-1b"
 },
 "Monitoring": {
   "State": "disabled"
 },
 "SubnetId": "subnet-03e0cc288b1bf9e67",
 "VpcId": "vpc-0473ebd347ec8b538",
 "PrivateIpAddress": "172.31.6.33"
}

PS C:\WINDOWS\system32>
```

---

## Summary of Setup

- ✅ **VPC** (10.0.0.0/16) created.
- ✅ **Subnet** (10.0.1.0/24) created within the VPC.
- ✅ **Internet Gateway** created and attached to the VPC.
- ✅ **Route Table** created, associated with the subnet, and configured to allow internet access.