# Networking in AWS

Tamal Dey,

Dept. of CA, PESU

# Networking and Content Delivery

- **Amazon VPC**
- Amazon CloudFront
- Amazon Route 53
- AWS PrivateLink
- AWS Direct Connect
- AWS Global Accelerator
- Amazon API Gateway
- AWS Transit Gateway
- AWS App Mesh
- AWS Cloud Map
- Elastic Load Balancing

# Why VPC?

- Provision a logically isolated section of the Amazon Web Services (AWS) Cloud where you can launch AWS resources in a virtual network that you define.
- Creator has complete control over your virtual networking environment, including selection of your own IP address range, creation of subnets, and configuration of route tables and network gateways.
- You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications.
- You can easily customize the network configuration for your Amazon VPC.

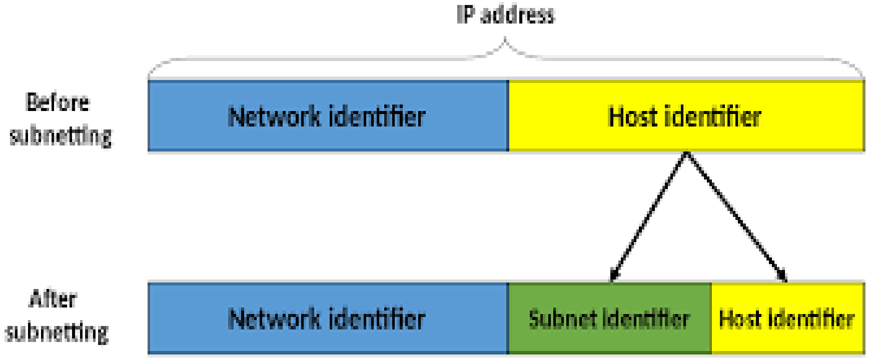
# What Is Amazon VPC?

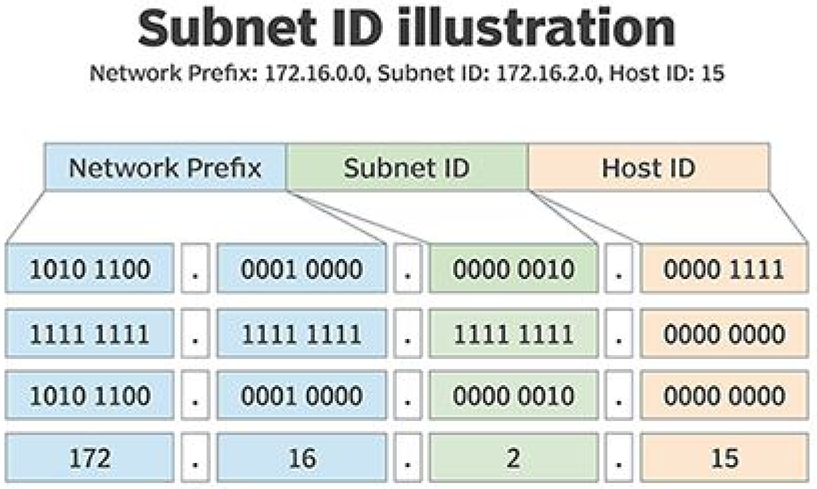
- Amazon Virtual Private Cloud (Amazon VPC) enables you to launch AWS resources into a virtual network that you've defined. This virtual network closely resembles a traditional network that you'd operate in your own data center, with the benefits of using the scalable infrastructure of AWS.
- **Benefits**
  - Define Custom Networks.
  - Assign static private IPv4 address to instances.
  - Define network interface and attach one or more network interface to the instances.
  - Define the routing between different subnets.
  - Define network security  by allowing or denying the traffic.
  - Control the out bound traffic along with inbound traffic using ACL.

# VPC Key Concepts

- A ***subnet*** is a range of IP addresses in your VPC. To launch AWS resources into a specified subnet. Use a public subnet for resources that must be connected to the internet, and a private subnet for resources that won't be connected to the internet.
- To protect the AWS resources in each subnet, you can use multiple layers of security, including **security groups** and **network access control lists** (**ACL**).
- A virtual private cloud (VPC) is a virtual network dedicated to your AWS account. It is logically isolated from other virtual networks in the AWS Cloud. You can launch your AWS resources, such as Amazon EC2 instances, into your VPC.
- VPC in the form of a **Classless Inter-Domain Routing** (CIDR) block; for example, ***10.0.0.0/16.*** This is the primary CIDR block for your VPC, a way to allow more flexible allocation of Internet Protocol (IP) addresses.

# Subnet



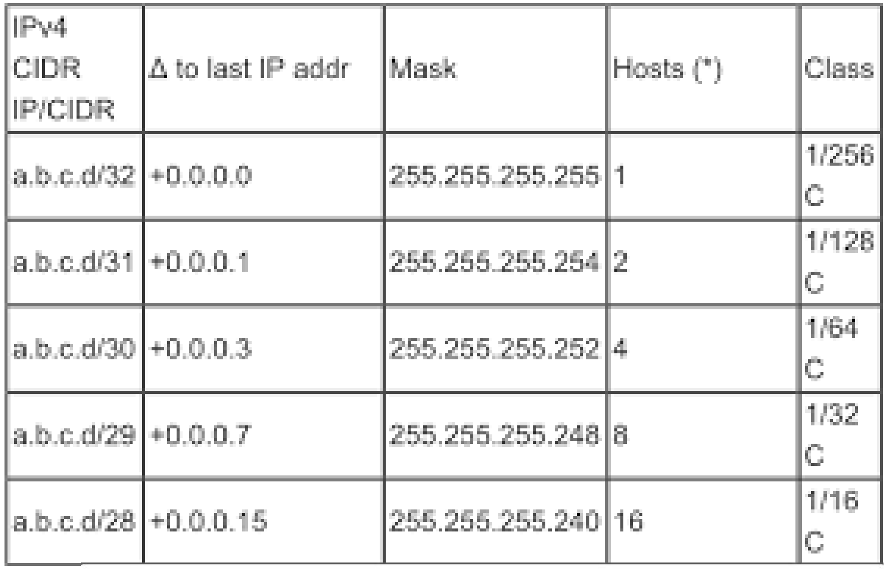


## Subnet ID illustration

Network Prefix: 172.16.0.0, Subnet ID: 172.16.2.0, Host ID: 15

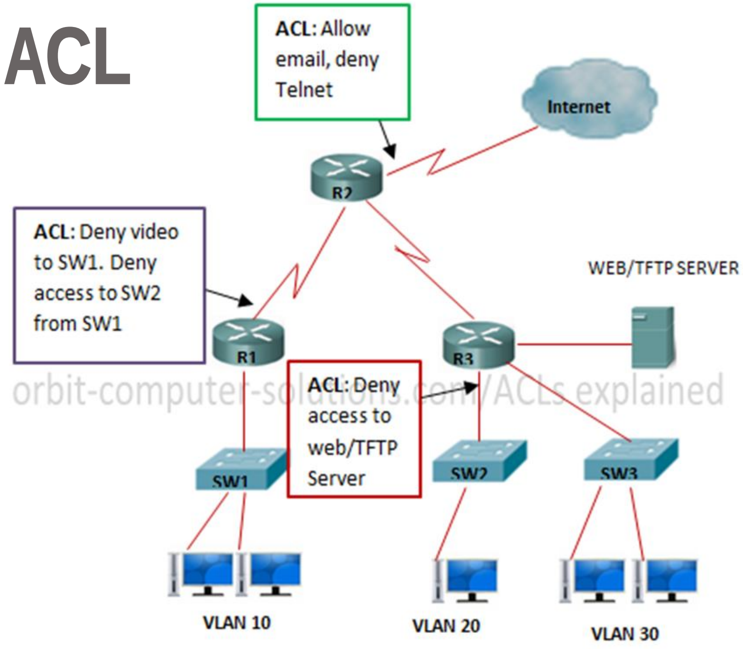
| Network Prefix | | | Subnet ID | | Host ID |
|---|---|---|---|---|---|
| 1010 1100 | . 0001 0000 | . | 0000 0010 | . | 0000 1111 |
| 1111 1111 | . 1111 1111 | . | 1111 1111 | . | 0000 0000 |
| 1010 1100 | . 0001 0000 | . | 0000 0010 | . | 0000 0000 |
| 172 | . 16 | . | 2 | . | 15 |

# CIDR



| IPv4<br>CIDR<br>IP/CIDR | Δ to last IP addr | Mask | Hosts (*) | Class |
|---|---|---|---|---|
| a.b.c.d/32 | +0.0.0.0 | 255.255.255.255 | 1 | 1/256<br>C |
| a.b.c.d/31 | +0.0.0.1 | 255.255.255.254 | 2 | 1/128<br>C |
| a.b.c.d/30 | +0.0.0.3 | 255.255.255.252 | 4 | 1/64<br>C |
| a.b.c.d/29 | +0.0.0.7 | 255.255.255.248 | 8 | 1/32<br>C |
| a.b.c.d/28 | +0.0.0.15 | 255.255.255.240 | 16 | 1/16<br>C |

# ACL


ACL: Allow email, deny Telnet
Internet
R2
ACL: Deny video to SW1. Deny access to SW2 from SW1
WEB/TFTP SERVER
R1
R3
ACL: Deny access to web/TFTP Server
SW1
SW2
SW3
VLAN 10
VLAN 20
VLAN 30

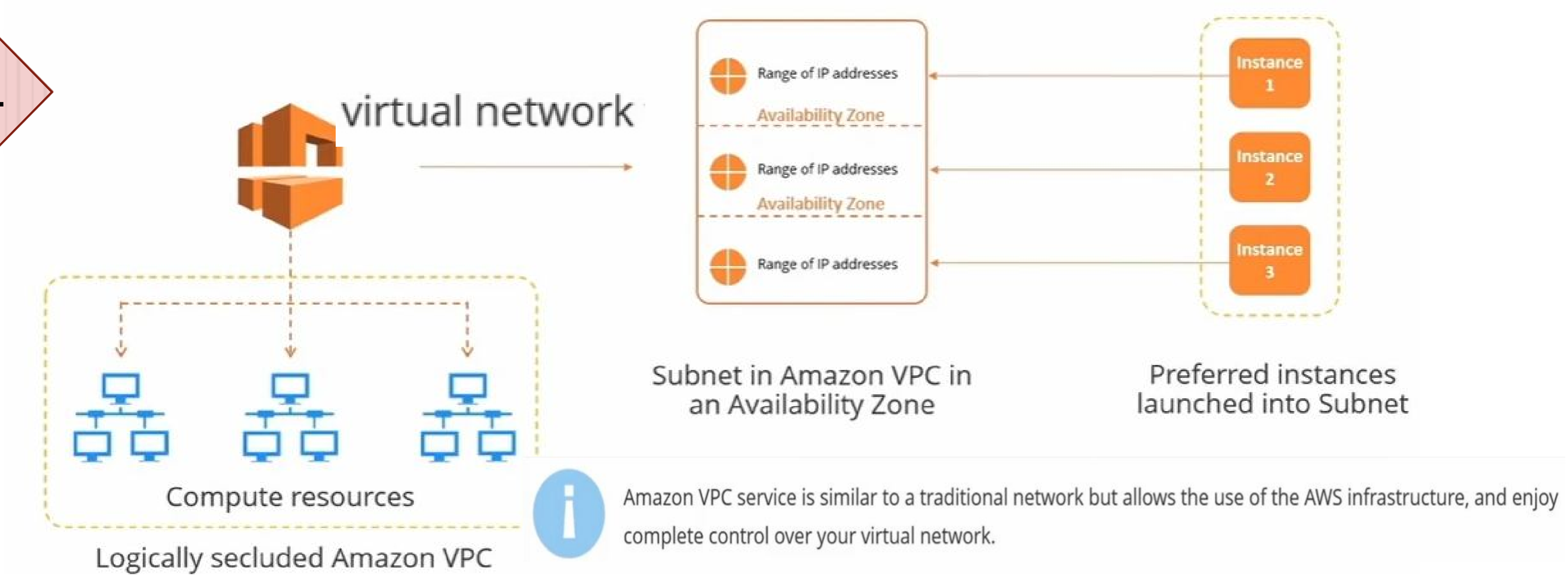
orbit-computer-solutions.com/ACLs explained

# VPC Key Concepts

- **Security Groups**: Each rule is comprised of four fields: 'Type', 'Protocol', 'Port Range', and 'Source'. This applies for both 'Inbound' and 'Outbound' rules.
- The drop down list allows you to select common protocols like SSH, RDP, or HTTP
- A **network access control list** *(**ACL)*** is an optional layer of security for your VPC that acts as a firewall for controlling traffic in and out of one or more subnets.
- A **route table** contains a set of rules, called *routes*, that are used to determine where network traffic is directed.
- Each subnet in your VPC must be associated with a route table; the table controls the routing for the subnet. A subnet can only be associated with one route table at a time, but you can associate multiple subnets with the same route table.
- An **internet gateway** is a horizontally scaled, redundant, and highly available VPC component that allows communication between instances in your VPC and the **internet.**

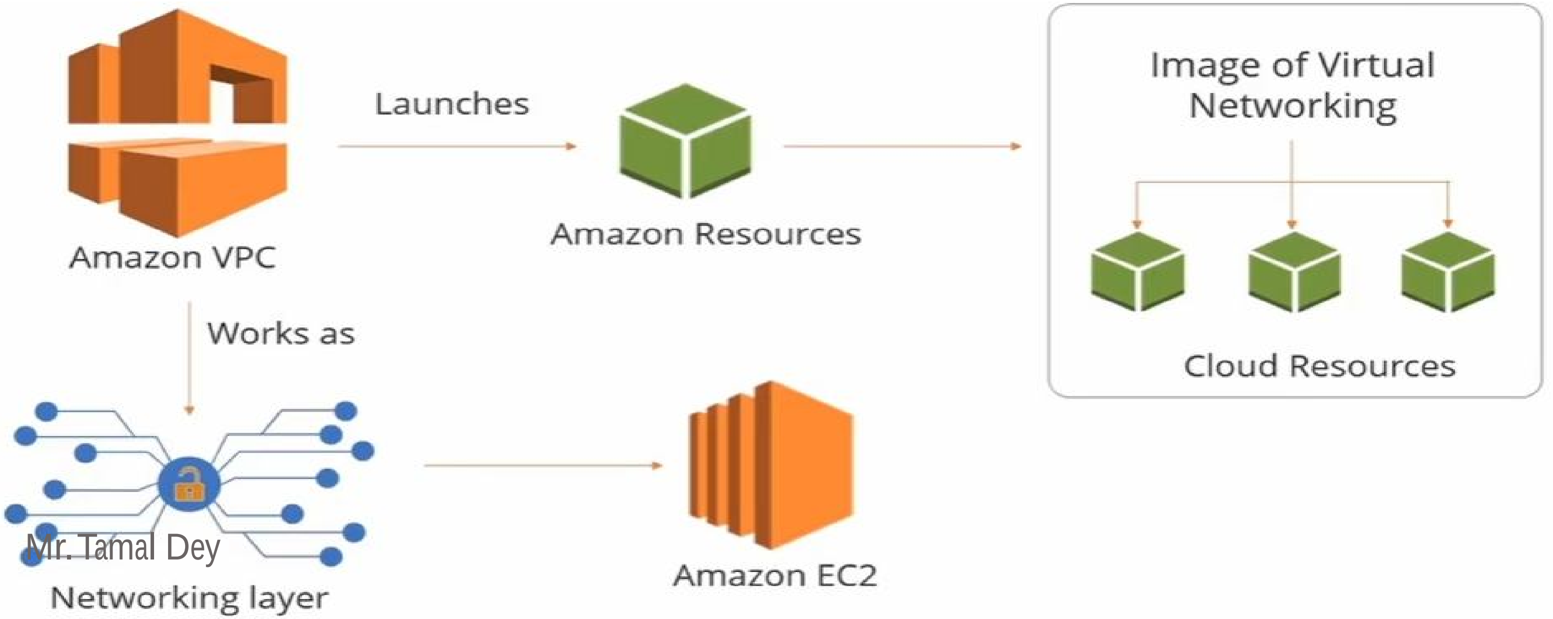
# Amazon Virtual Private Cloud

1

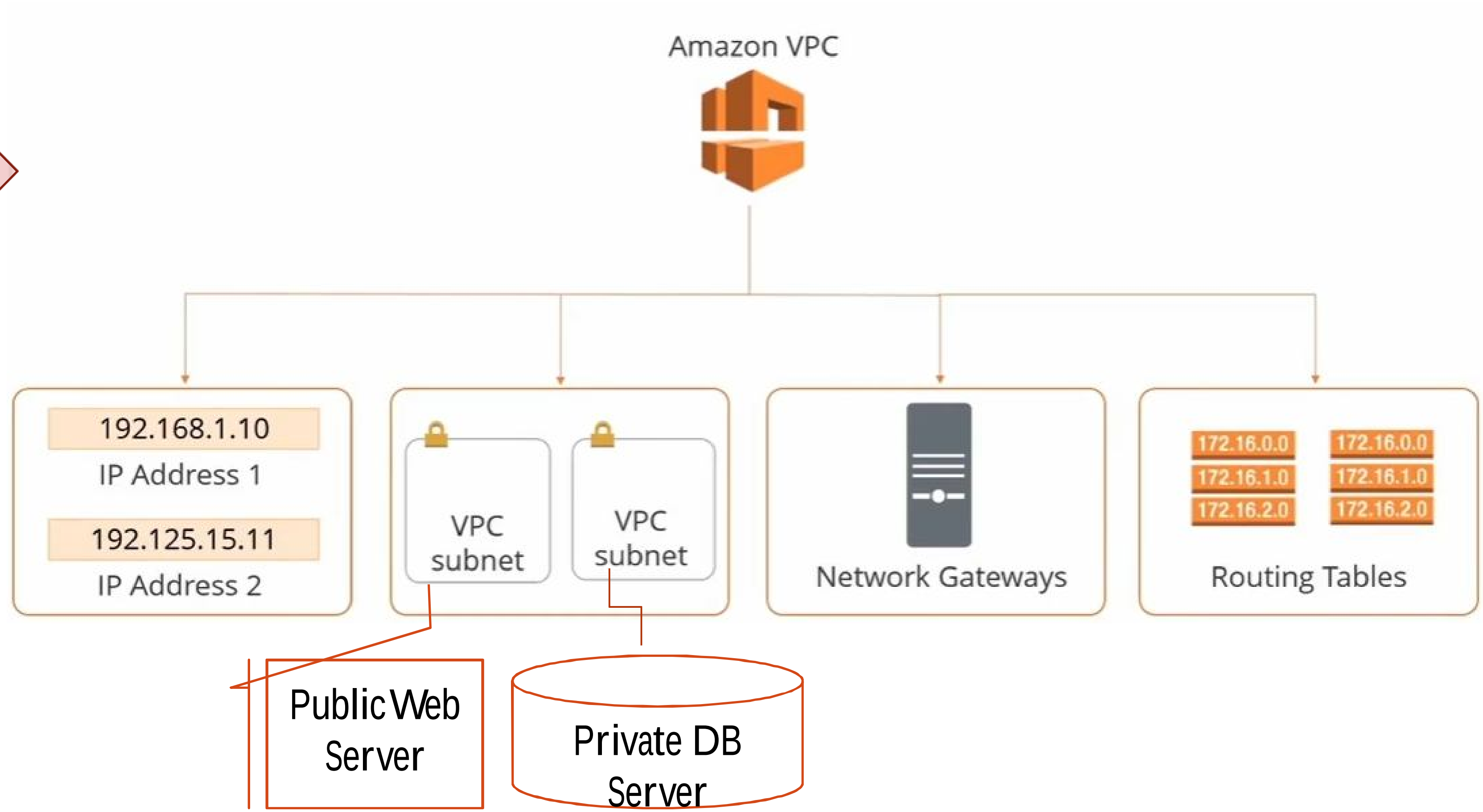


Amazon VPC service is similar to a traditional network but allows the use of the AWS infrastructure, and enjoy complete control over your virtual network.

2



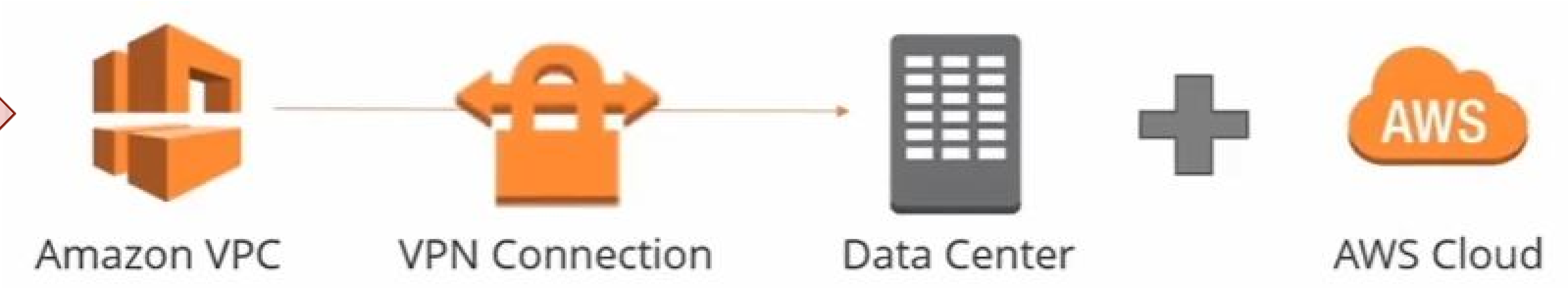
3



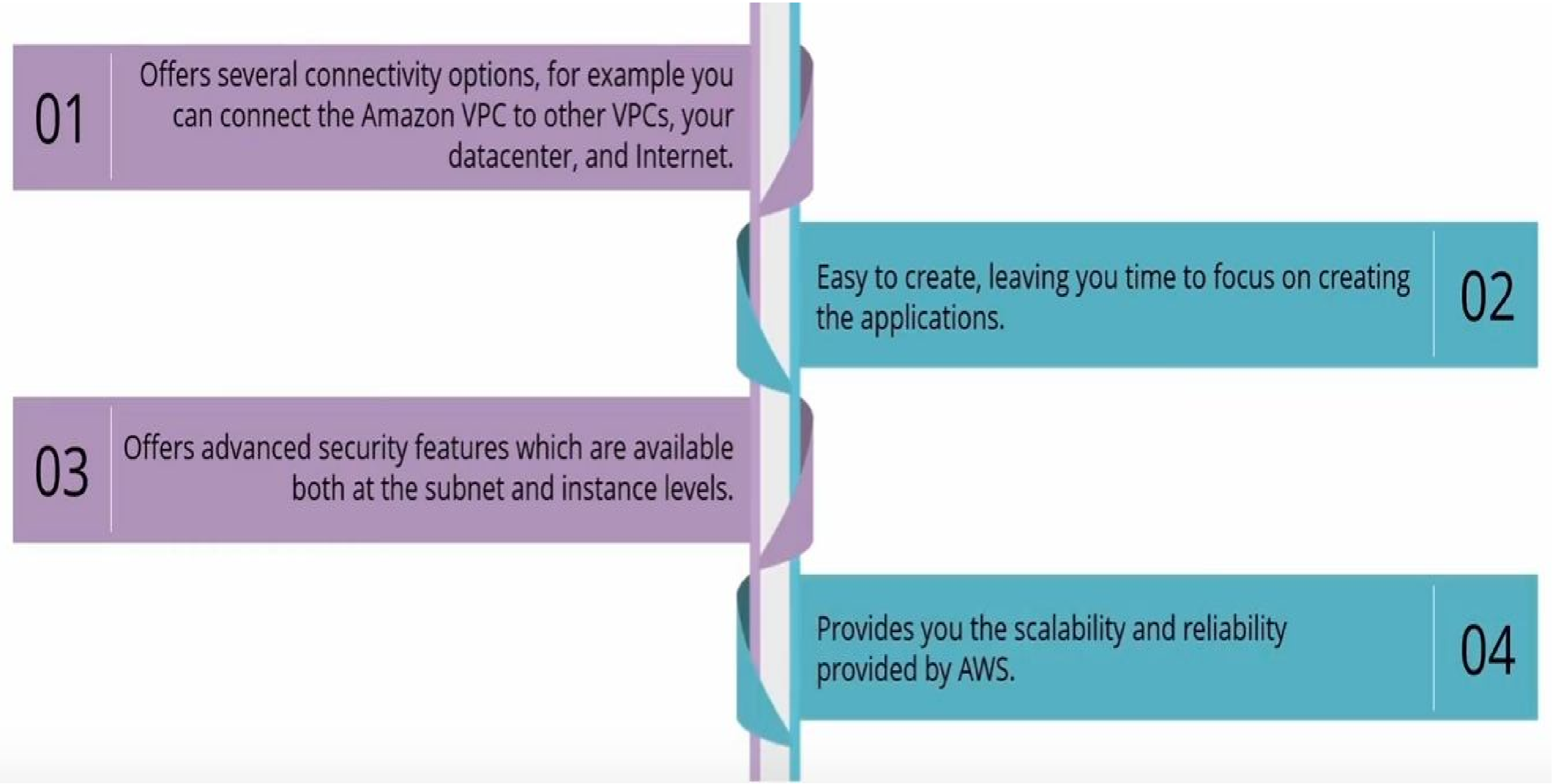
Public Web Server

Private DB Server

4



# Benefits of AWS



01 Offers several connectivity options, for example you can connect the Amazon VPC to other VPCs, your datacenter, and Internet.

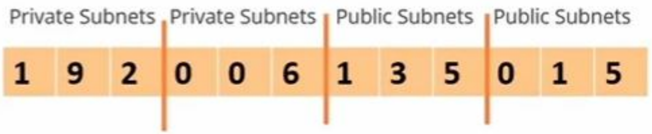
Easy to create, leaving you time to focus on creating the applications. 02

03 Offers advanced security features which are available both at the subnet and instance levels.

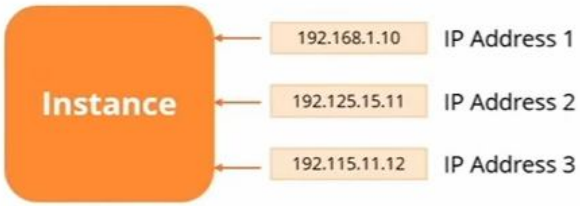
Provides you the scalability and reliability provided by AWS. 04

# Benefits of Launching Instances in VPC

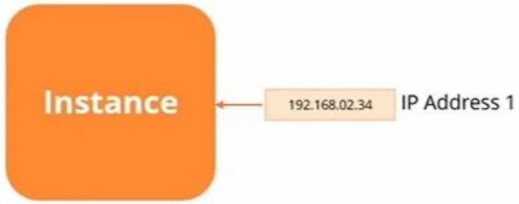
Run Instances on the hardware used by a single entity



Split the range of private IP addresses of VPC



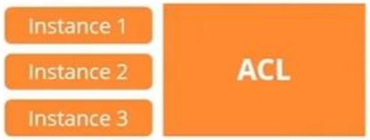
Allocate multiple IP addresses to Instances



Allocate static private IP addresses to Instances

Define Network Interfaces

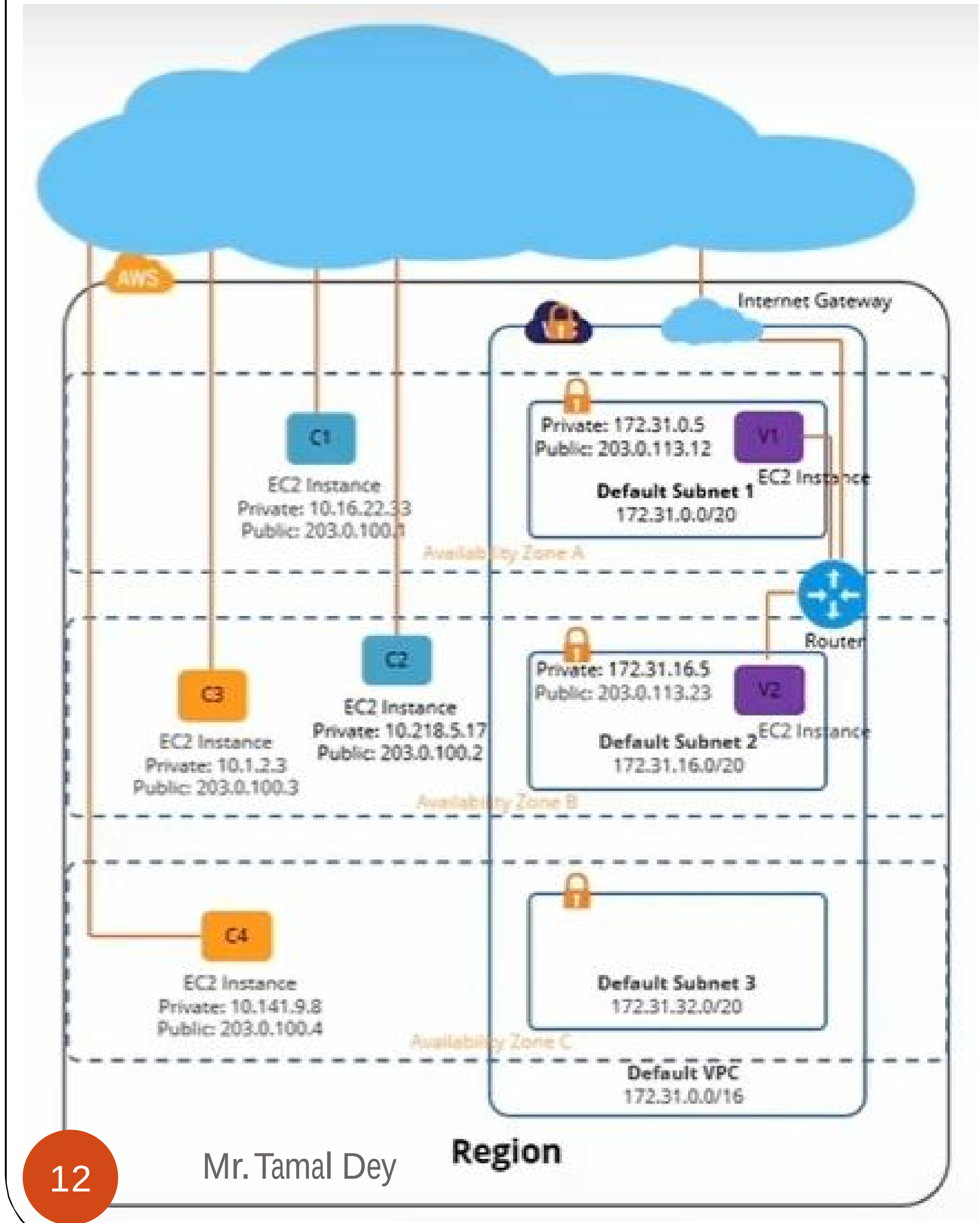
Control inbound traffic to Instances



Add an extra layer of Access Control to Instances

Change the membership of Security Group of Instances

# Default and Non-Default VPC



The default VPC contains a Subnet in each availability zone.

It is ready to use, offering advanced features of the EC2-VPC platform.

Even with an AWS account, you can create and configure a VPC as per your requirements.

Additional subnets in a default VPC and a non-default VPC are termed as non-default Subnets.

SCENARIO

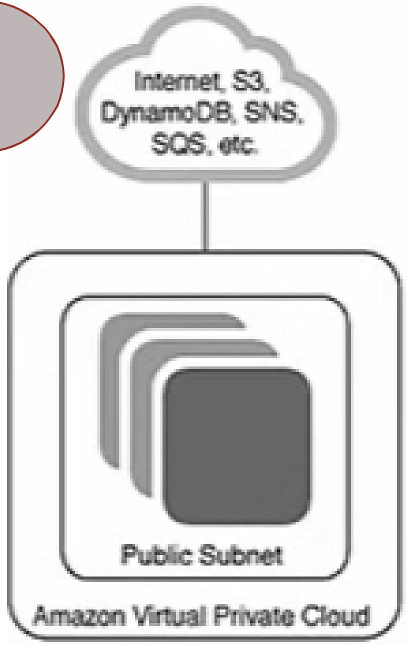
VPC with a Single Public Subnet

VPC with Public and Private Subnets

VPC with Public and Private Subnets and Hardware VPN Access

VPC with a Private Subnet Only and Hardware VPN Access

1

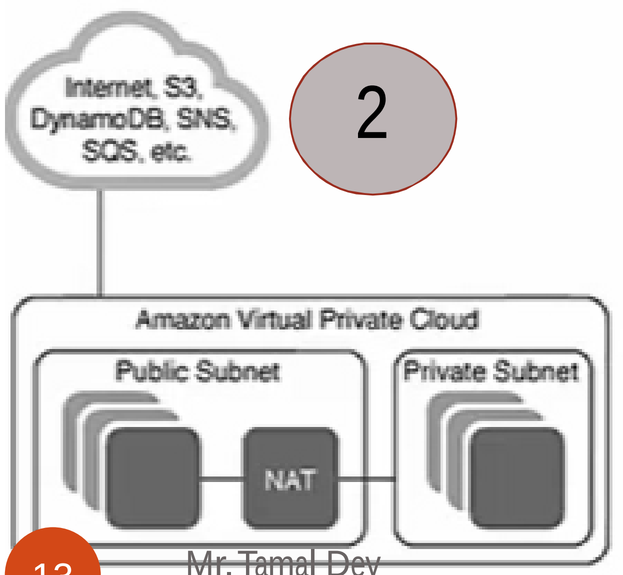

Internet, S3, DynamoDB, SNS, SQS, etc.
Public Subnet
Amazon Virtual Private Cloud

**VPC with a Single Public Subnet:** **Here i**nstances run in a private, isolated section of the AWS cloud with direct access to the Internet. Network access control lists and security groups can be used to provide strict control over inbound and outbound network traffic to your instances.

**Creates:**

A /16 network with a /24 subnet. Public subnet instances use Elastic IPs or Public IPs to access the Internet.

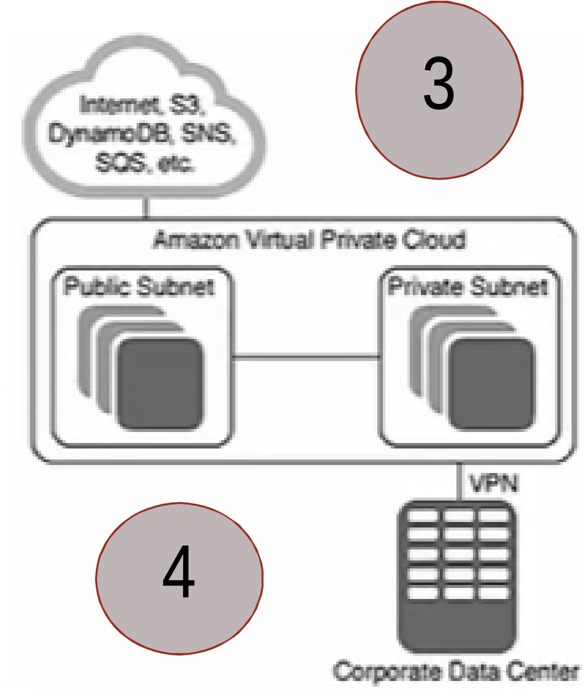
2


Internet, S3, DynamoDB, SNS, SQS, etc.
Amazon Virtual Private Cloud
Public Subnet
Private Subnet
NAT

**VPC with Public and Private Subnets: A** public subnet, this configuration adds a private subnet whose instances are not addressable from the Internet. Instances in the private subnet can establish outbound connections to the Internet via the public subnet using Network Address Translation (NAT).

**Creates:**

A /16 network with two /24 subnets. Public subnet instances use Elastic IPs to access the Internet. Private subnet instances access the Internet via Network Address Translation (NAT). (Hourly charges for NAT devices apply.)


Internet, S3, DynamoDB, SNS, SQS, etc.
Amazon Virtual Private Cloud
Public Subnet
Private Subnet
VPN
Corporate Data Center

3

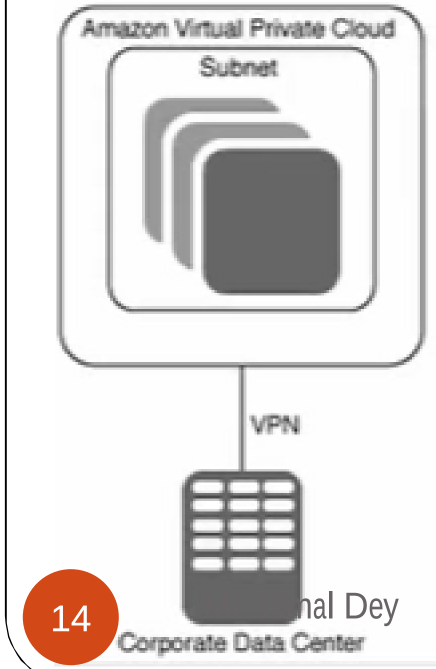
## VPC with Public and Private Subnets and Hardware VPN Access

This configuration adds an IPsec Virtual Private Network (VPN) connection between your Amazon VPC and your data center - effectively extending your data center to the cloud while also providing direct access to the Internet for public subnet instances in your Amazon VPC.

**Creates:**

A /16 network with two /24 subnets. One subnet is directly connected to the Internet while the other subnet is connected to your corporate network via IPsec VPN tunnel. (VPN charges apply.)

4



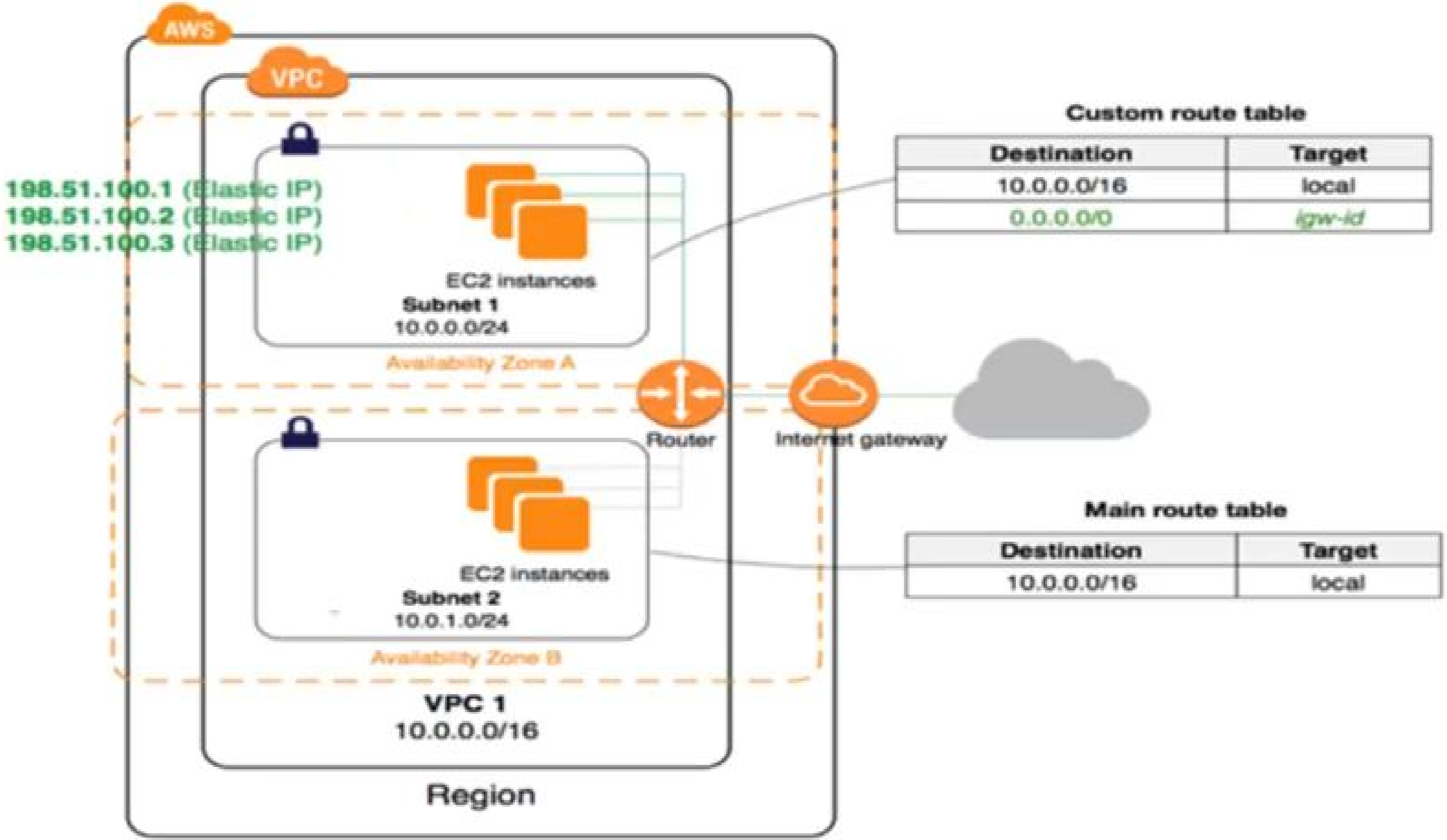
## VPC with Private Subnets Only and Hardware VPN Access

our instances run in a private, isolated section of the AWS cloud with a private subnet whose instances are not addressable from the Internet. You can connect this private subnet to your corporate data center via an IPsec Virtual Private Network (VPN) tunnel.

**Creates:**

A /16 network with a /24 subnet and provisions an IPsec VPN tunnel between your Amazon VPC and your corporate network. (VPN charges apply.)

# Implementation Scenario



Custom route table

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 0.0.0.0/0 | *igw-id* |

Main route table

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |

# Steps to Follow

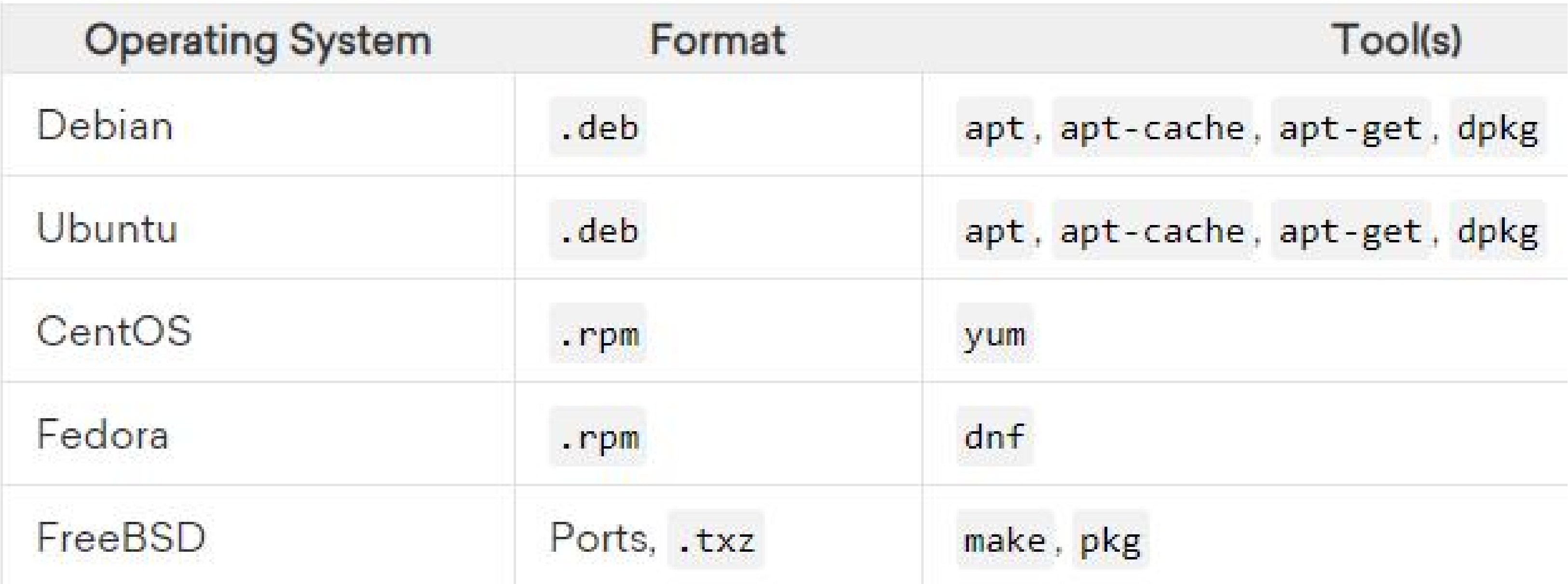
1. Network->VPC
   - A. Check/Note Default VPC, Subnet, Internet Gateway, Routing Table, Network ACLs, Security Options, DHCP Option Set
2. Create new VPC (MyVPC)
   - A. IPV4 CIDR Block range (10.0.0.0/16) with default Tenancy
   - B. Check all the fields of new VPC
   - C. Check Route Table, Network ACLs, Security Groups
   - D. No Subnet Created and Internet Gateway
3. Create 2 Subnet Name(10.0.1.0-AP S 1A & 10.0.2.0-AP S 1B) in New custom VPC (1p-s-1a and 1b)
   - A. IPv4 CDIR Block- 10.0.1.0/24 and 10.0.2.0/24
   - B. 1 Subnet in 1 Availability zone . Check new subnet details
4. Create Internet Gateway(MyInternetGateway)
   - A. Attach to VPC [ 1 Gateway only for 1VPC]

5. Create Route Table(MyPublicRoute) from subnet ->Internet Gateway

   A. Edit (save) / Add new Route (0.0.0.0/0) and Target (MyInternetGateway)

   B. Subnet Association (Public -1A Subnet-> Internet Gateway)

   C. Go to Subnet and (Right side> auto assign public IPv4 address enable from Actions button- Auto-assign IPv4



| Operating System | Format | Tool(s) |
|---|---|---|
| Debian | `.deb` | `apt`, `apt-cache`, `apt-get`, `dpkg` |
| Ubuntu | `.deb` | `apt`, `apt-cache`, `apt-get`, `dpkg` |
| CentOS | `.rpm` | `yum` |
| Fedora | `.rpm` | `dnf` |
| FreeBSD | Ports, `.txz` | `make`, `pkg` |

6. Launce One EC2 instance in Public Subnet

   A. Amazon AMI (Free Tier) , change to new VPC and Public subnet)
   B. Advance Details [Add the **text** in next slide]

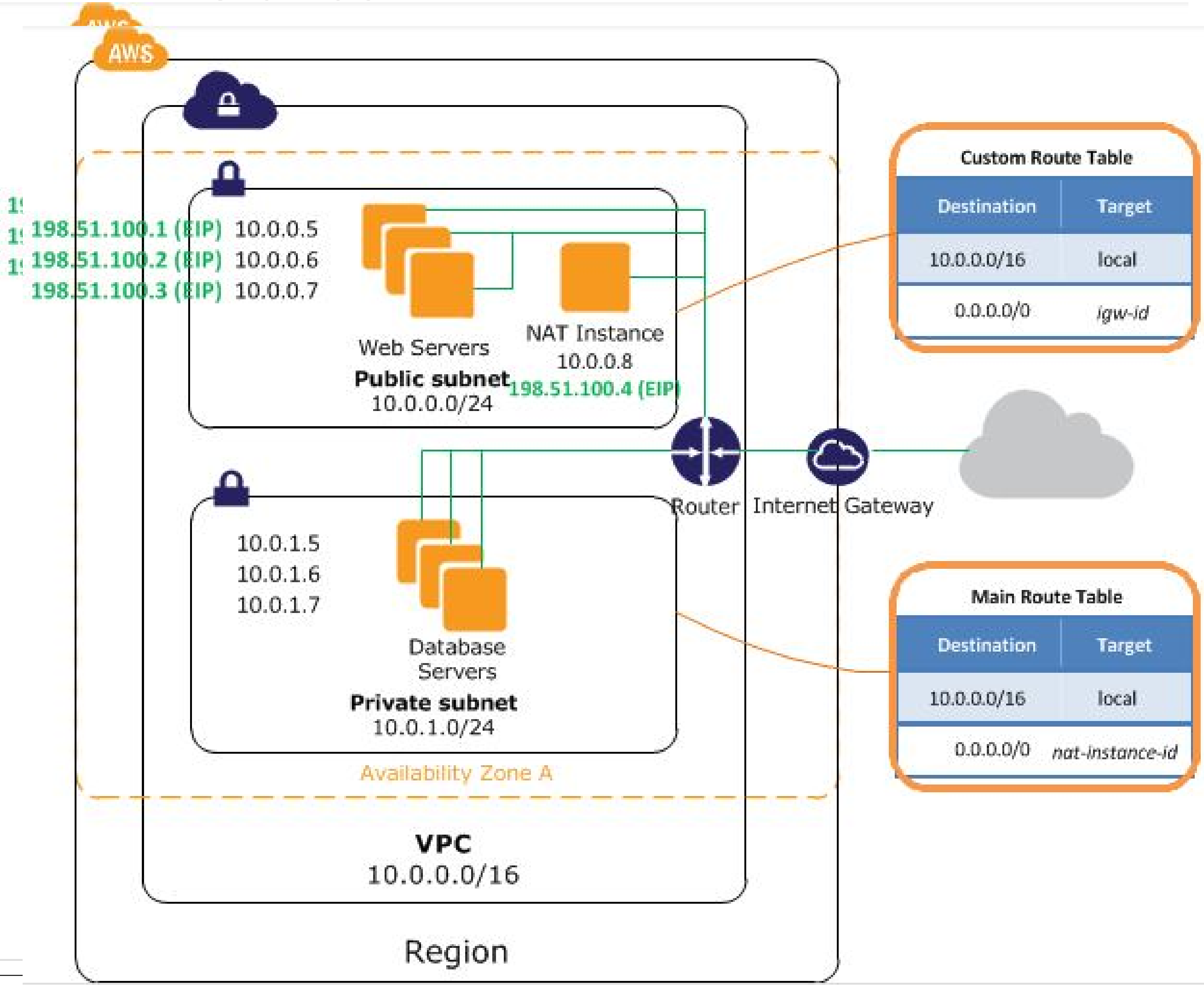
```
#!bin/bash
yum install httpd -y
yum update -y
service httpd start
chkconfig httpd on
echo "<html><h1>Hello World!</h1>
</html>" > /var/www/html/index.html
```

   C. Add default Storage and Add Tag (Name -> Webserver)
   D. Security group Name and Desc. (WebSec) and Add HTTP Protocol
   E. Download new private key pair (WebSec.pem) & Launch Instance
   F. Write auto created IPv4 address in browser (Result)

7. Launce One EC2 instance Private instance
   A. Create Amazon AMI (Free Tier) instance
   B. Choose Custom VPC and Private Subnet with no auto assign IP address
   C. Add storage and Add Tag (Name-> DBServer) [Private Access]
   D. Add Security group protocol for the following (with **Public-IP Address**)
      I. SSH – 10.0.1.0/24
      II. MySQL-10.0.1.0/24
      III. ICMP-IPv4- 10.0.1.0/24
      IV. ICMP-IPv4- 10.0.1.0/24
   E. Download the DB Key pair (WebSec.pem-Existing key pair)
   F. Run with Putty (Windows User) by using private key
      I. Ping DBServer IP Address

End of VPC Type 1

# NAT Instance



Custom Route Table

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 0.0.0.0/0 | *igw-id* |

Main Route Table

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 0.0.0.0/0 | *nat-instance-id* |

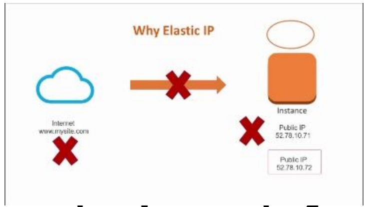
# NAT Instances

- **Network Address Translation** (NAT) instance in a public subnet in your VPC to enable instances in the private subnet to initiate outbound IPv4 traffic to the Internet or other AWS services, but prevent the instances from receiving inbound traffic initiated by someone on the Internet.
- *NAT is not supported for IPv6 traffic*—use an egress-only Internet gateway.
- ***Note:*** Use a NAT gateway, which is a managed NAT service that provides better availability, higher bandwidth, and requires less administrative effort.
- For common use cases, we recommend that you use a *NAT gateway* rather than a *NAT instance*.

# NAT Gateways

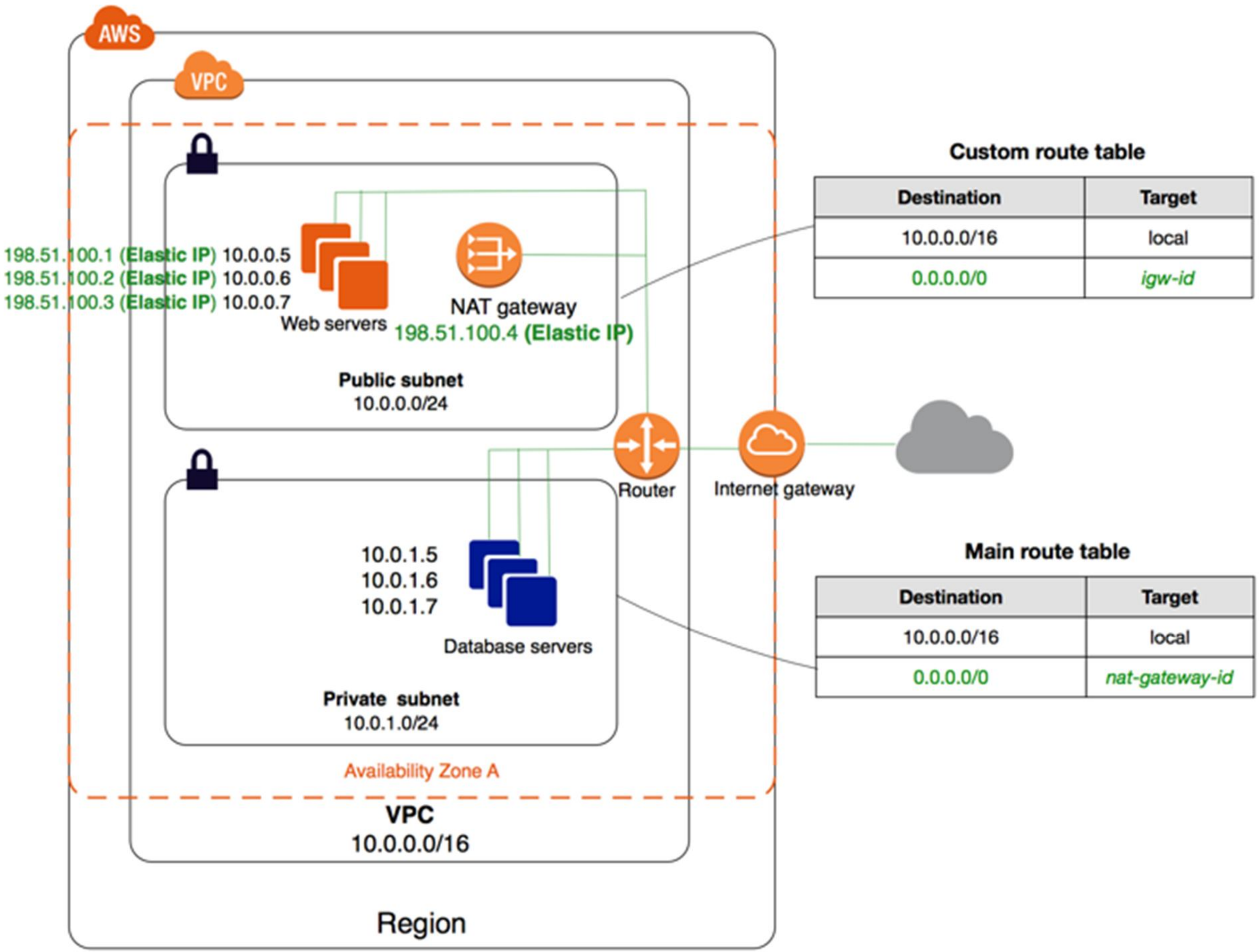
- Enable instances in a private subnet to connect to the internet or other AWS services, but prevent the internet from initiating a connection with those instances.
- You are charged for creating and using a NAT gateway in your account. NAT gateway hourly usage and data processing rates apply. Amazon EC2 charges for data transfer also apply.
  - https://aws.amazon.com/vpc/pricing/
- NAT gateways are not supported for IPv6 traffic
- Diff. NAT Instance vs. NAT Gateway
- https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat-comparison.html

# Elastic IP



- An **Elastic IP** address is a static IPv4 address designed for dynamic cloud computing.
- An **Elastic IP** address is associated with your AWS account.
- With an **Elastic IP** address, you can mask the failure of an instance or software by rapidly remapping the address to another instance in your account.
- An Elastic IP address is a public IPv4 address, which is reachable from the internet.
- If your instance does not have a public IPv4 address, you can associate an Elastic IP address with your instance to enable communication with the internet;
  - for example, to connect to your instance from your local computer.

# Lab 6-NAT Gateway



Custom route table

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 0.0.0.0/0 | *igw-id* |

Main route table

| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 0.0.0.0/0 | *nat-gateway-id* |

# Steps for NAT Gateway

1. Go to EC2 Instance
   A. **Search** in Community Instance (Left Side menu)
      i. NAT (First Option-Free Tier Only)
      ii. Choose Custom VPC and Public Subnet
      iii. All other default Setting, Add a Tag - NATInstance
      iv. Choose existing Security Group created for Public Subnet (WebServer)
      v. Choose Existing Key Pair (WebSec.pem) for key generation
      vi. Launch the instance
      vii. Select your NAT instance after Launch- and Go to

      Action ->Networking-> Change Source/ Destination change (Bypassing the request in this Point)

B. Go to VPC and Select NAT Gateways-> Create One
   i. Choose **public subnet** and create New Elastic IP
   ii. Create NAT Gateway
   iii. Edit Rout Table -> Select default subnet route
   iv. Edit (save) / Add new Route (0.0.0.0/0) and Target (NAT Gateway)

2. To Check the working of the NAT Gateway
   A. Login to Public subnet (putty)
   B. Transfer WebSec.pem to public subnet (WinSCP)
   C. chmod 400 WebSec.pem
   D. ssh -i "WebSec.pem" ec2-user@Private Server-IP
   E. ping google (8.8.8.8)

## Custom VPC Deletion

- No Running instances (Under Custom VPC)
- Go to  NAT Gateway –> Action ->Delete First
- Go to VPC -> Action -> Delete Custom VPC

## Default VPC Deletion and Create again

- No Running instances (under Default VPC )
- Go to VPC -> Action -> Delete Default VPC and Create same