iNurture
Education Solutions
Tomorrow's Here

# AWS Networking-VPC

TOMORROW'S HERE

# Topics to be covered

- Introduction of Network
- Public and Private Cloud
- Amazon VPC
- Subnets
- Internet gateway
- Route table
- Elastic IP address
- NAT gateway
- Elastic network interface

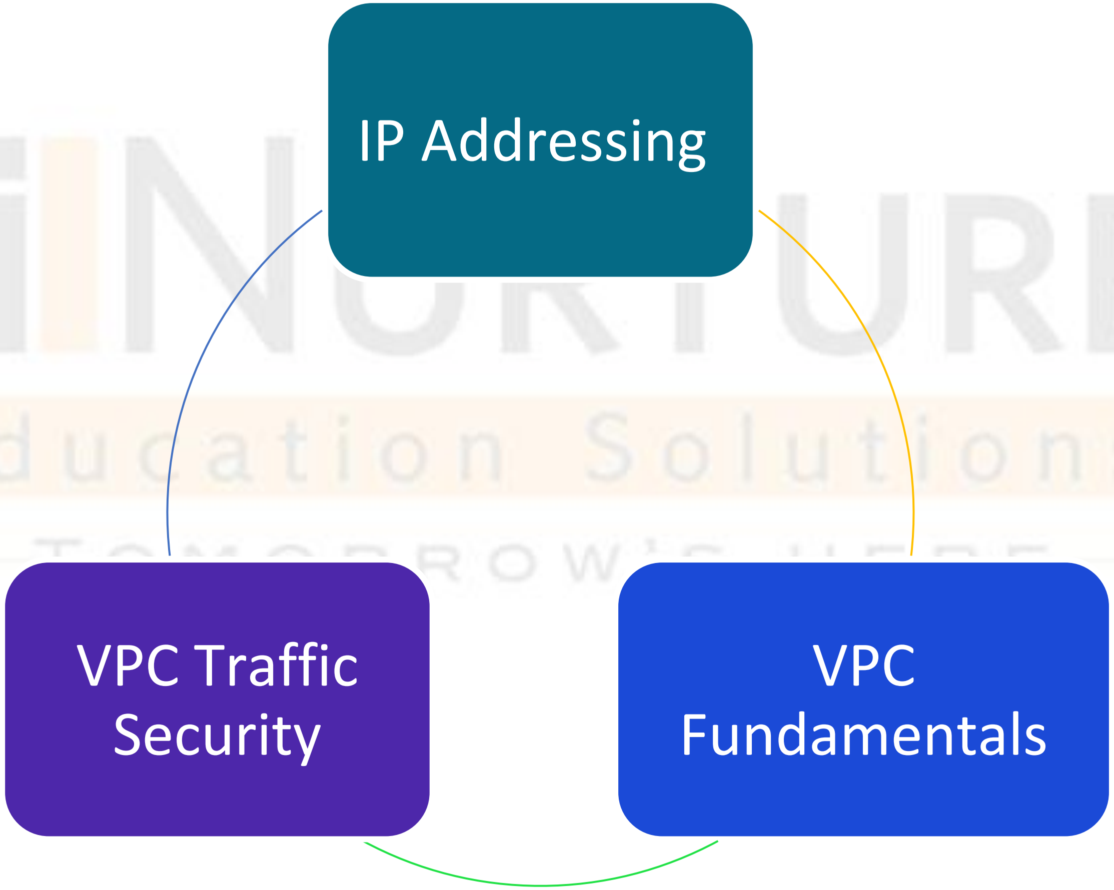
# Objective

How do I build a dynamic and secure network infrastructure in my AWS accounts?

How can I divide my private cloud into subnets and assign IP addresses?

How can I direct and filter traffic moving in and out of a network?

# Solutions



# IP Address

- IPv4 was developed in the early 1980s and uses 32-bit addresses.
- Numbers are grouped in fours, giving eight groups or blocks (four octets).
- Addresses in IPv4 are written using numeric dot-decimal notation.
- IPv4 allows for 4.3 billion addresses, meaning addresses must be reused and masked.
- IPv4 uses numeric dot-decimal notation.
  - Example: 192.168.5.18

iiNURTURE Education Solutions TOMORROW'S HERE

# IP Address

- IPv6 was developed in 1998 and uses 128-bit addresses.
- Numbers are grouped in fours, giving eight groups or blocks (16 octets).
- The groups are written with a colon as a separator.
- The addresses allow for 340 trillion addresses, meaning devices can have a unique address.
- IPv6 uses alphanumeric hexadecimal notation.
    - Example: Simplified – 50b2:6400::6c3a:b17d:0:10a9

iNURTURE Education Solutions TOMORROW'S HERE

# CIDR

- You specify this set of addresses in the form of a CIDR block—for example, 10.0.0.0/16.

| | |
|---|---|
| 0.0.0.0/0 | = All IPs |
| 10.22.33.44/32 | = 10.22.33.44 |
| 10.22.33.0/24 | = 10.22.33.* |
| 10.22.0.0/16 | = 10.22.*.* |

| CIDR | Total IPs |
|---|---|
| /28 | 16 |
| … | … |
| /20 | 4,096 |
| /19 | 8,192 |
| /18 | 16,384 |
| /17 | 32,768 |
| /16 | 65,536 |

# CIDR

- Amazon VPC supports IPv4 and IPv6 addressing, and it has different CIDR block size limits for each.
- By default, all VPCs and subnets must have IPv4 CIDR blocks—you can't change this behavior, but you can optionally associate an IPv6 CIDR block with your VPC.

iNURTURE Education Solutions TOMORROW'S HERE

# Public Vs Private IP Address

- A public IP address identifies you to the wider internet.
- A private IP address strengthens network security.

## Public IP Address

Public IP: 54.56.9.10

Public IP addresses are IPv4 addresses reachable from the internet.

## Private IP Address

Private IP: 172.31.1.90

Private IP addresses are not reachable from the internet.

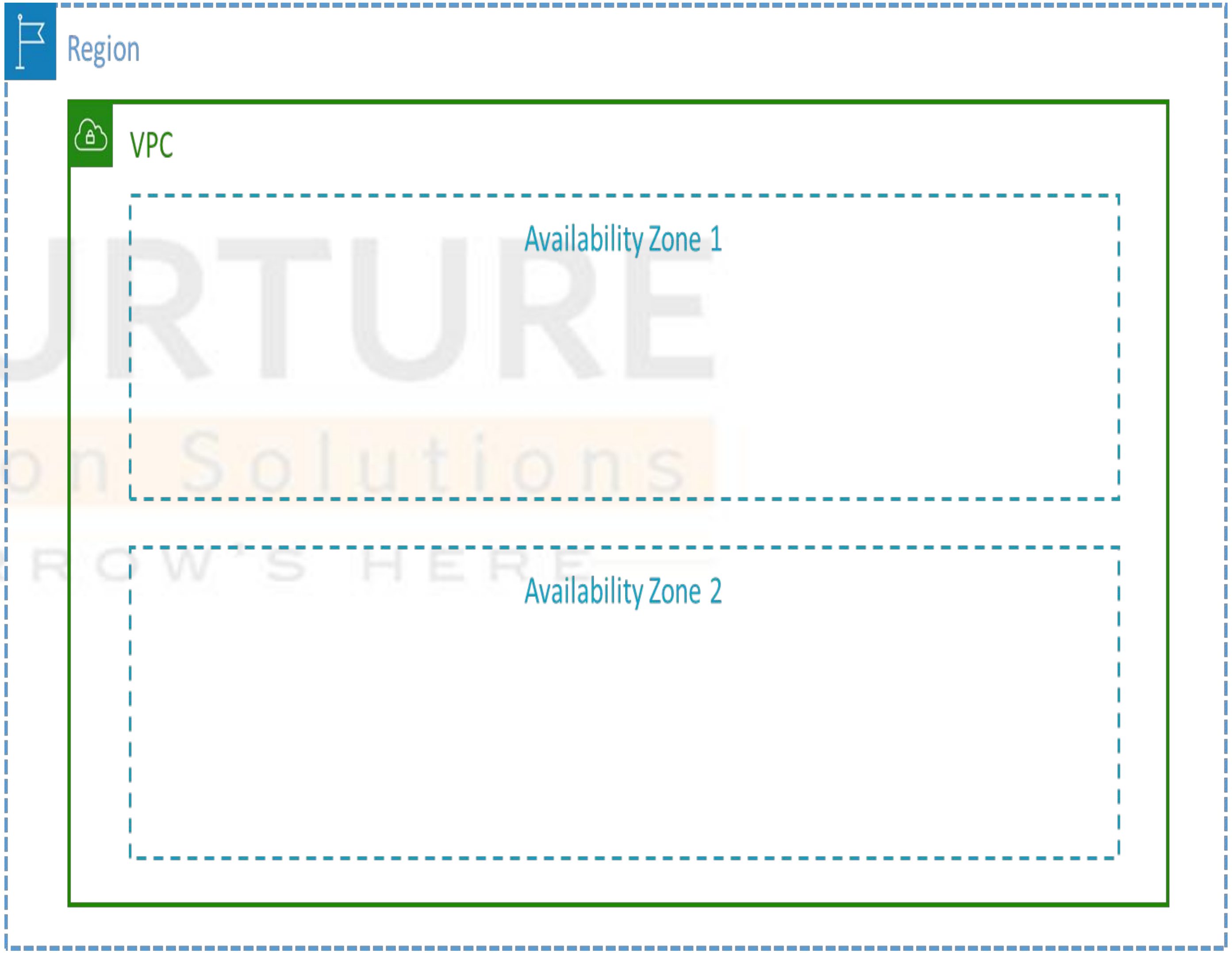
- ❖ You can use private IP addresses for communication between instances in the same VPC.
- ❖ A private IP address is used inside your private network.
- ❖ Every EC2 instance in a VPC has at least one private IP address.
- ❖ An EC2 instance can also be assigned a public IP address.

# VPC

- A virtual private cloud (VPC) is a secure, isolated private cloud hosted within a public cloud.
- VPC customers can run code, store data, host websites, and do anything else they could do in an ordinary private cloud, but the private cloud is hosted remotely by a public cloud provider.
- VPC is a Virtual Network or Data Center inside AWS for one client.
- It is  logically isolated from other Virtual Networks in AWS.
- Max 5 VPC can be created and 200 subnet in 1 VPC.
- We can allocate max 5 Elastic IP.

# VPC

- Once we created VPC
  - DHCP
  - NACL and
  - Security Group will be automatically created.
- A VPC is confined to an AWS Region and does not extend between regions.



iNurture Education Solutions TOMORROW'S HERE

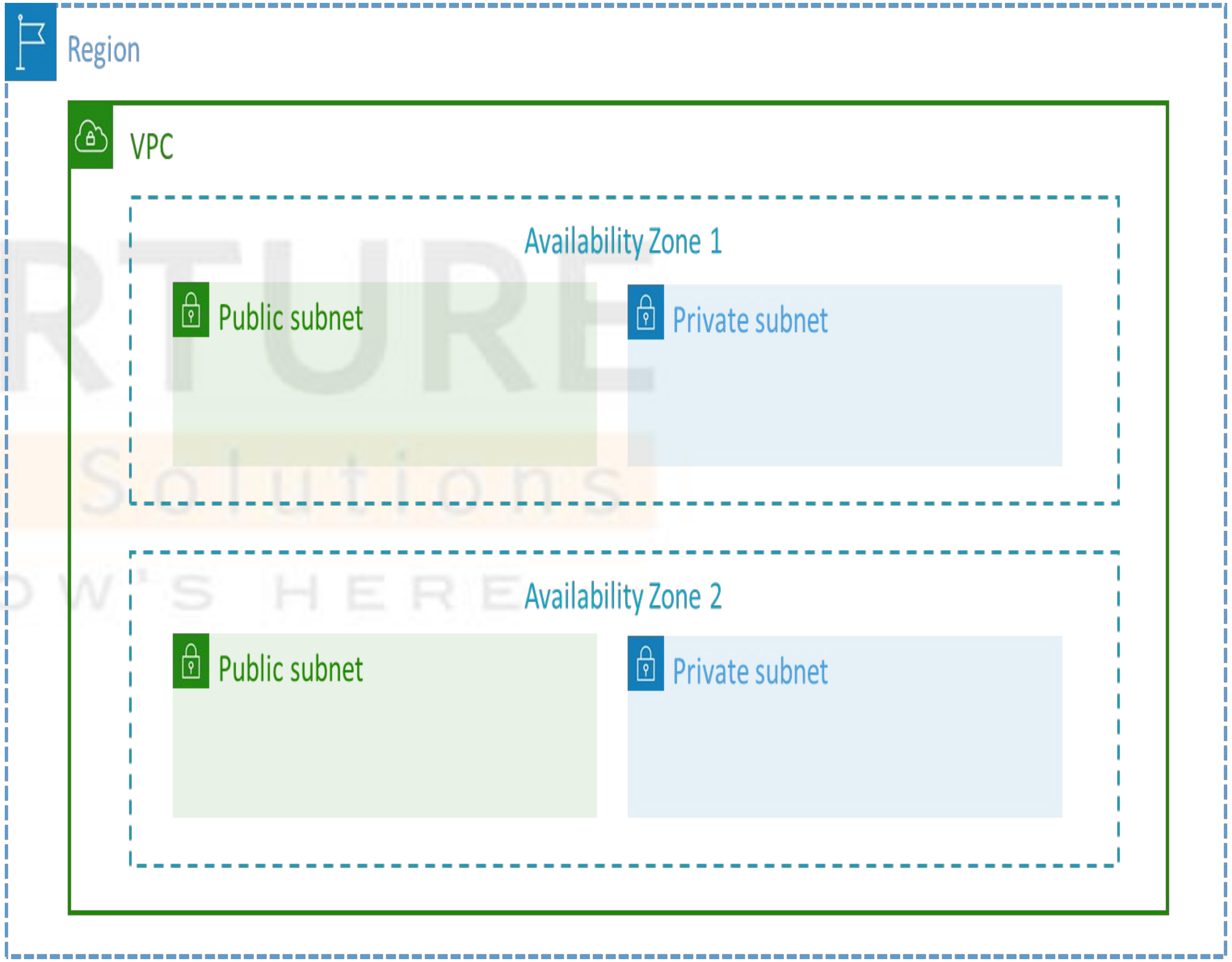
# VPC

- Amazon VPC is designed to provide greater control over the isolation of your environments and their resources, including:
  - Selection of your own IP address range
  - Creation of subnets
  - Configuration of route tables and network gateways

iNURTURE Education Solutions TOMORROW'S HERE

# VPC

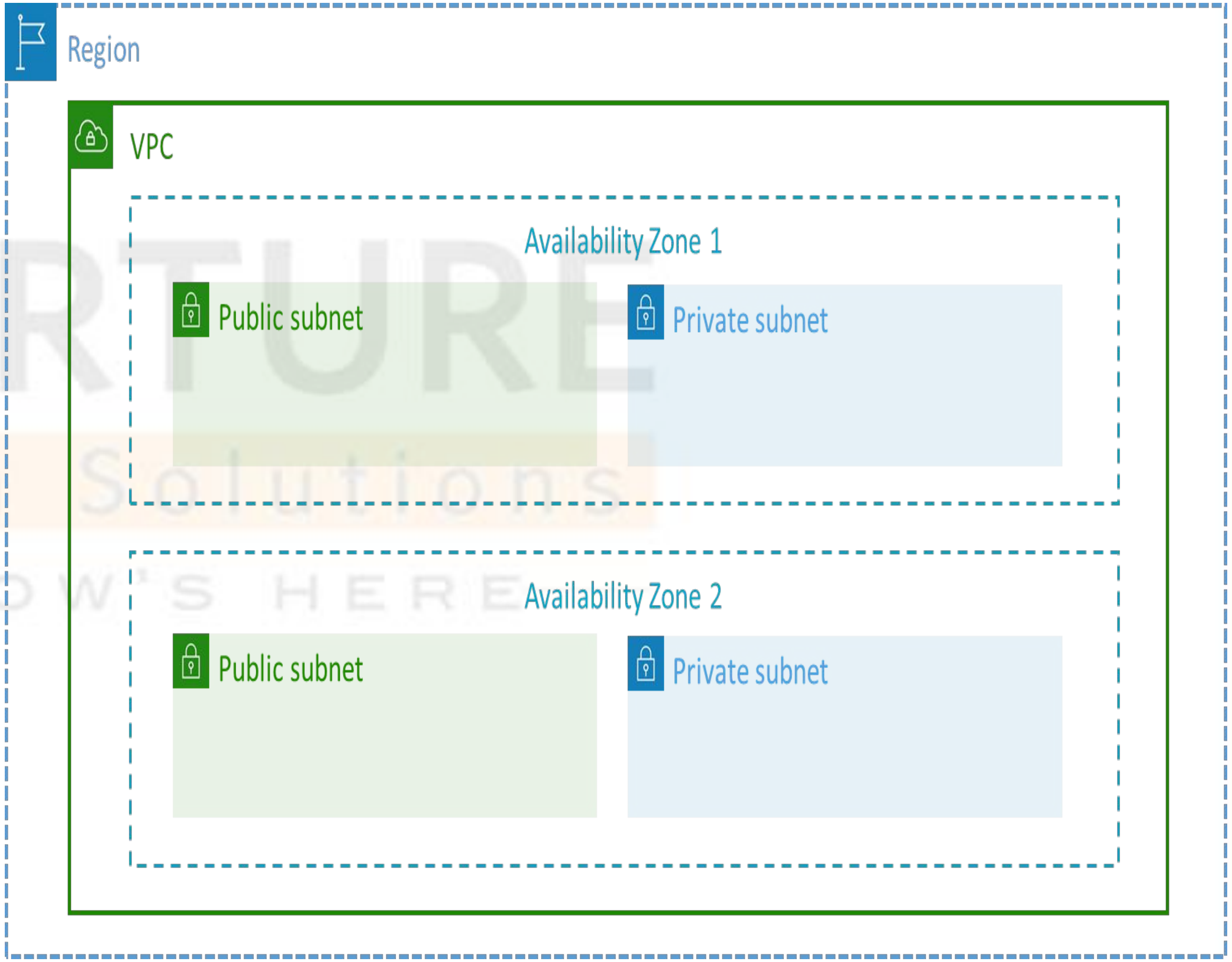
- A subnet is a range of IP addresses in your VPC.
- You can launch AWS resources into a specified subnet.
- Use a public subnet for resources that must be connected to the internet and a private subnet for resources that won't be connected to the internet.



iNURTURE Education Solutions TOMORROW'S HERE

# VPC

- A subnet resides within one Availability Zone.
- AWS reserves the first four IP addresses and the last IP address in each subnet CIDR block. Consider larger subnets over smaller ones (/24 and larger).
- You are less likely to waste or run out of IPs if you distribute your workload into larger subnets.



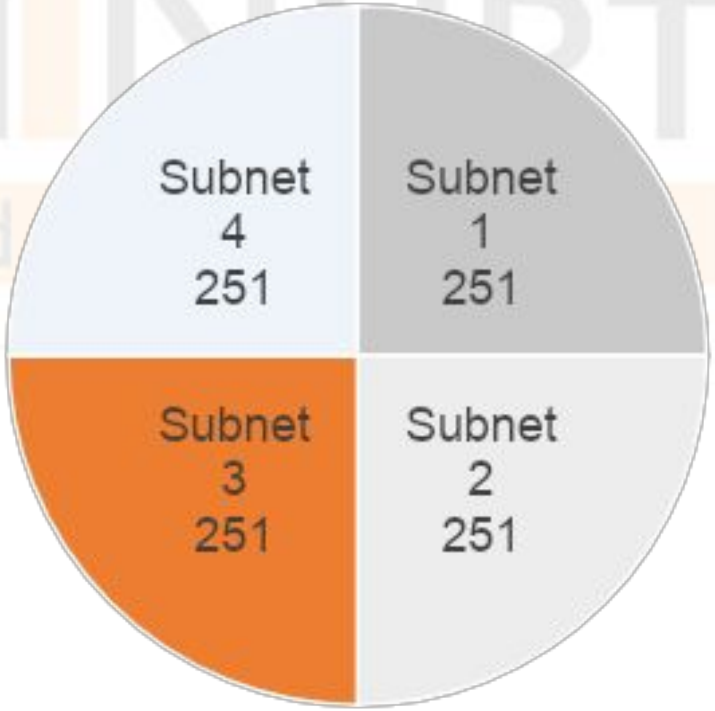
iNURTURE Education Solutions TOMORROW'S HERE

# Using Subnets to Divide Your VPC

- With Amazon VPC, customers can create virtual networks and divide them into subnets.
- VPC subnets are mapped to specific Availability Zones.
- You choose a CIDR block for the subnet, which is a subset of the VPC CIDR block.
- Each subnet must reside within one Availability Zone and cannot span zones.
- The first four IP addresses and the last IP address in each subnet CIDR block are not available and cannot be assigned to an instance.

# Using Subnets to Divide Your VPC

- Using subnets isolates resources for routing and security.
- AWS will reserve five IP addresses from each subnet.

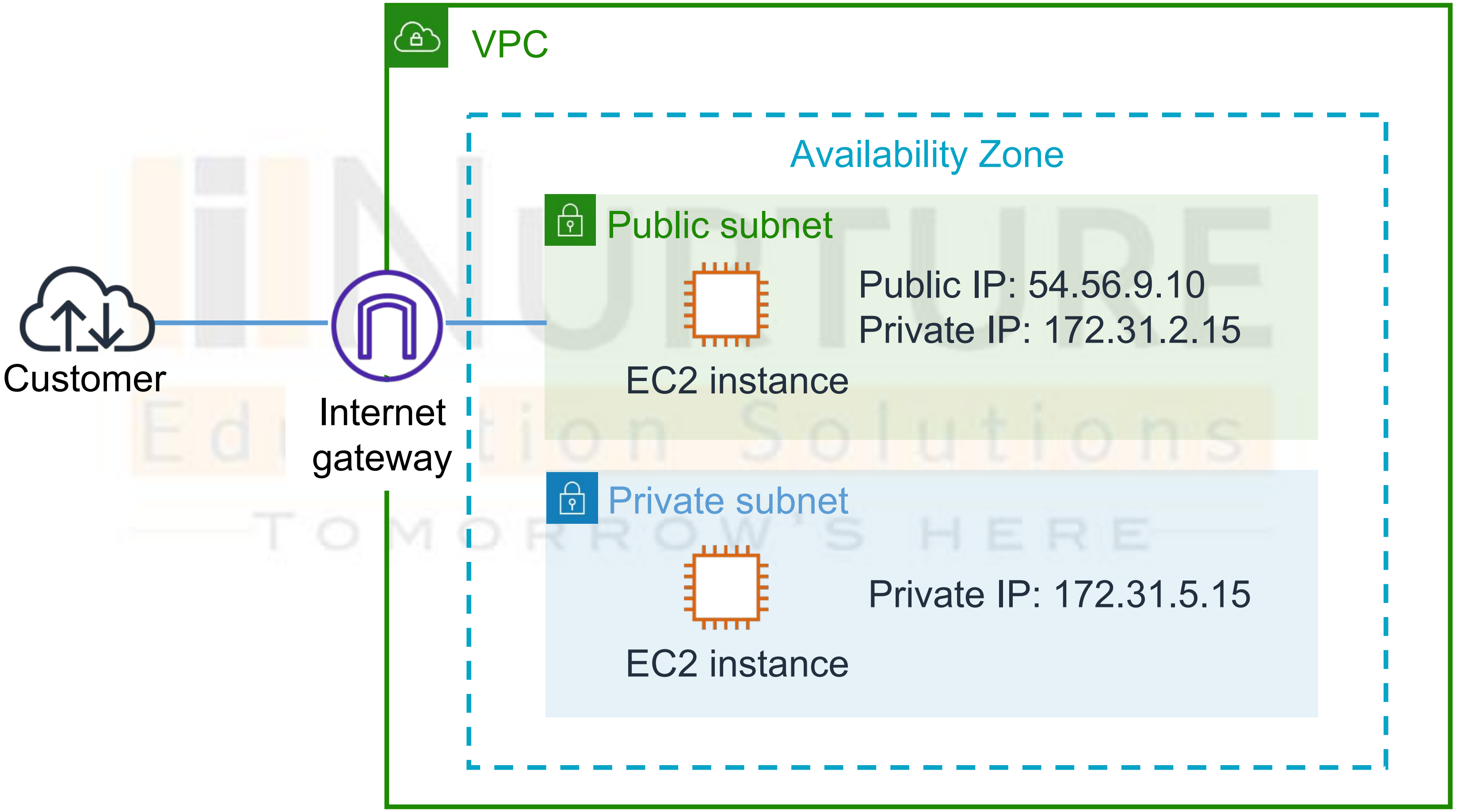


A VPC with **CIDR /22** includes 1,024 total IP addresses.

# Using Subnets to Divide Your VPC

- For example, in a subnet with CIDR block 10.0.0.0/24, the following five IP addresses are reserved:
  - 10.0.0.0: Network address.
  - 10.0.0.1: Reserved by AWS for the VPC router.
  - 10.0.0.2: Reserved by AWS.
    - The IP address of the DNS server is always the base of the VPC network range plus 2.
  - 10.0.0.3: Reserved by AWS for future use.
  - 10.0.0.255: Network broadcast address.
    - We do not support broadcast in a VPC; therefore, we reserve this address.

# Internet Gateway



iiNURTURE Education Solutions TOMORROW'S HERE

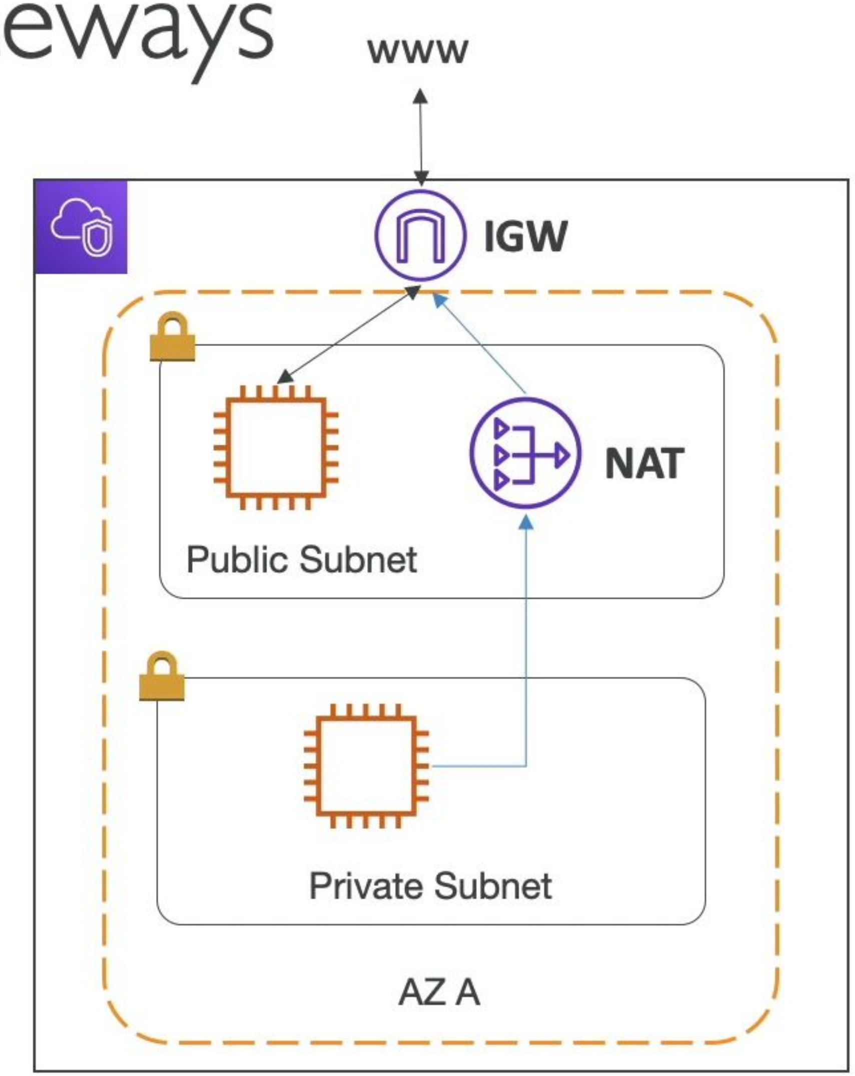
# Internet Gateway

- Internet gateways permit communication between instances in your VPC and the internet.
- They provide a target in your subnet route tables for internet-routable traffic.
- An internet gateway is a horizontally scaled, redundant, and highly available VPC component that permits communication between instances in your VPC and the internet.
- It imposes no availability risks or bandwidth constraints on your network traffic.
- An internet gateway serves two purposes:
  - To provide a target in your VPC route tables for internet-routable traffic
  - To perform network address translation (NAT) for instances that have been assigned public IPv4 addresses
- An internet gateway supports IPv4 and IPv6 traffic.

# Internet Gateway & NAT Gateways

- **Internet Gateways** helps our VPC instances connect with the internet
- Public Subnets have a route to the internet gateway.

- **NAT Gateways** (AWS-managed) & **NAT Instances** (self-managed) allow your instances in your **Private Subnets** to access the internet while remaining private

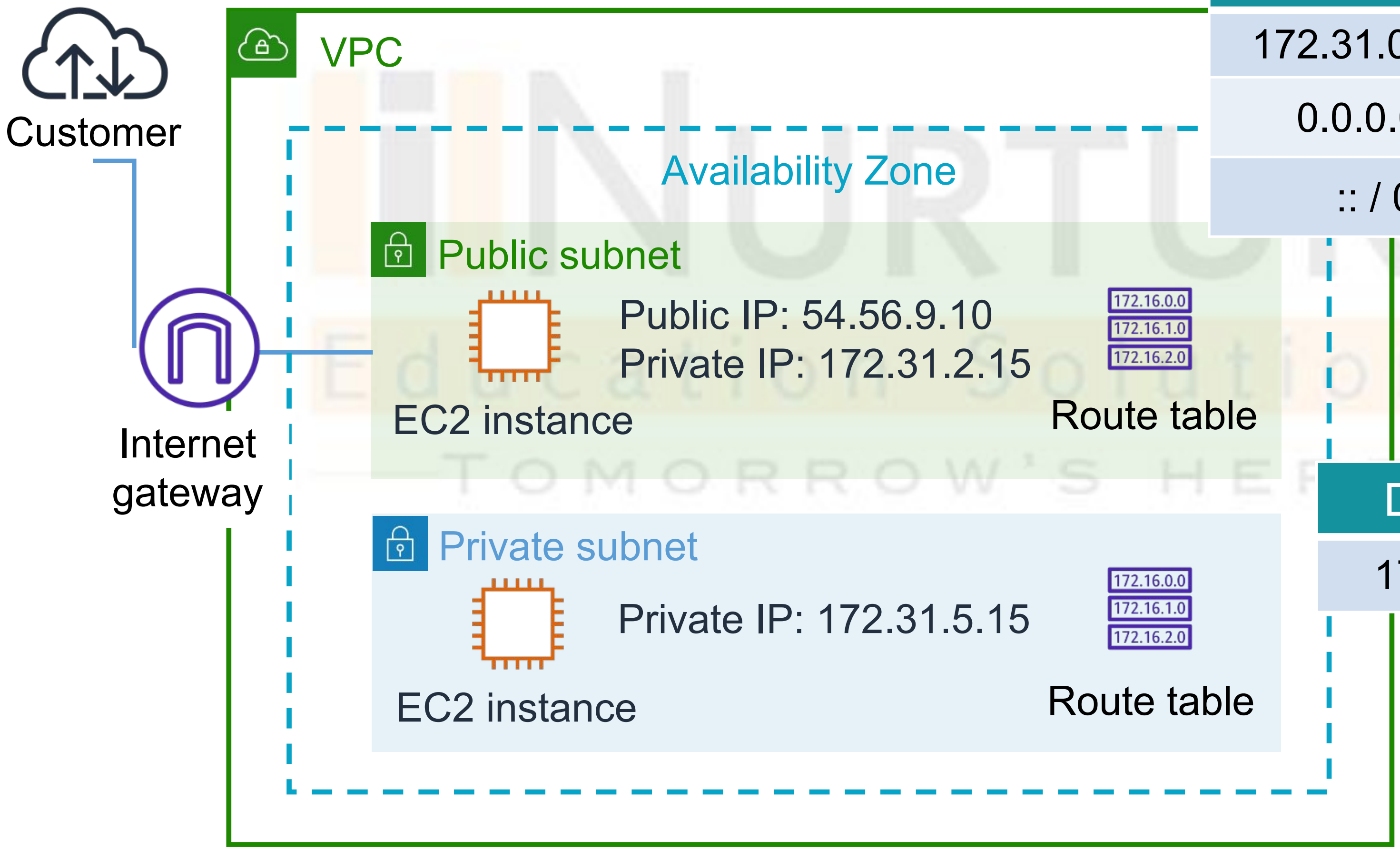


# Route Tables

- A route table contains a set of rules (routes) that are used to determine where network traffic is directed.
- When you create a VPC, it automatically has a main route table.
- Initially, the main route table (and every route table in a VPC) contains only a single route.
- This is a local route that permits communication for all the resources within the VPC.
- You can't modify the local route in a route table.
- You can create additional custom route tables for your VPC.

iNURTURE Education Solutions TOMORROW'S HERE

# Route Tables



Public route table

| Destination | Target |
|---|---|
| 172.31.0.0/16 | local |
| 0.0.0.0/0 | <igw-id> |
| :: / 0 | igw-xxx |

Private route table

| Destination | Target |
|---|---|
| 172.31.0.0/16 | local |

iNURTURE Education Solutions TOMORROW'S HERE

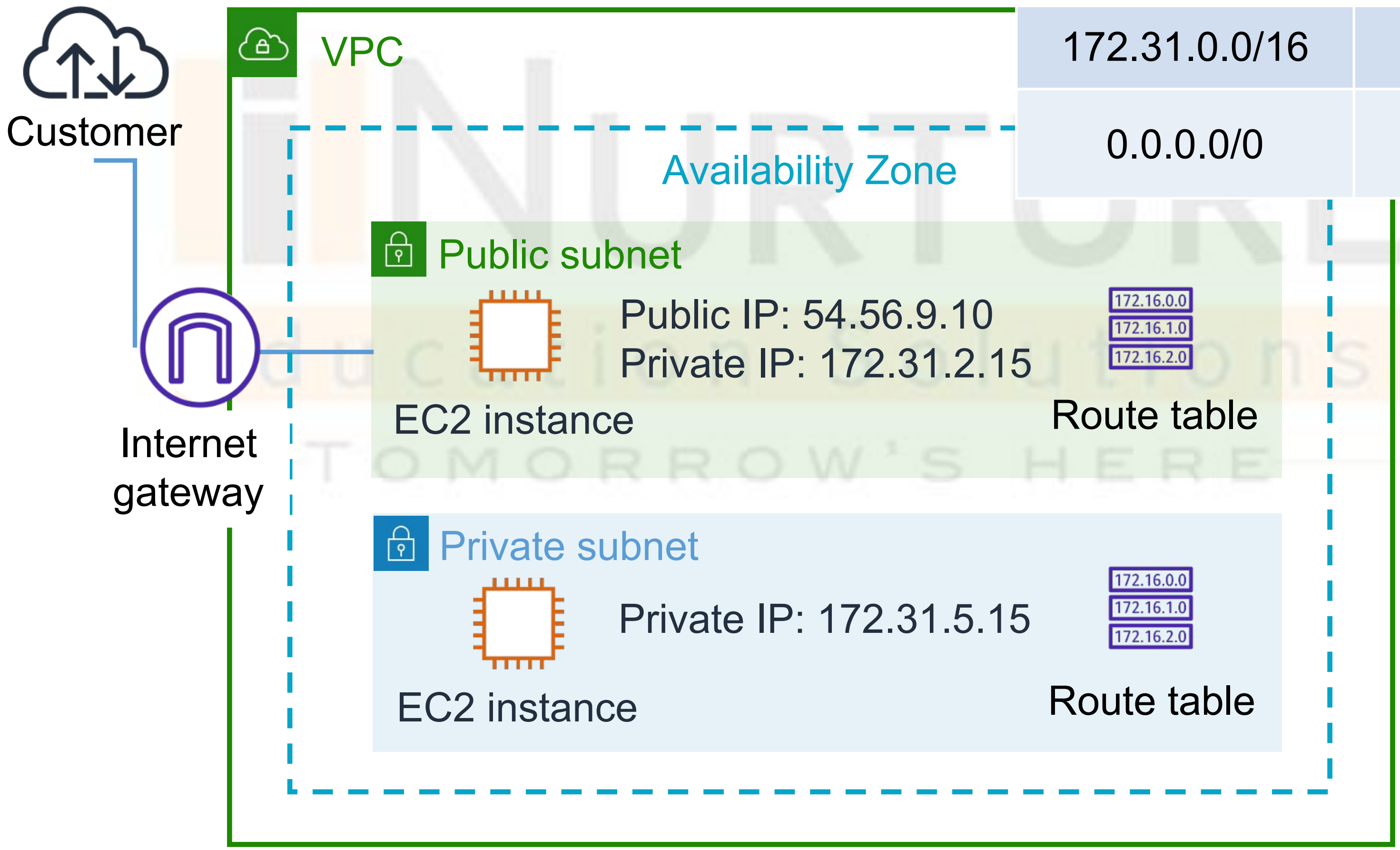
# Route Tables

- Each subnet in your VPC must be associated with a route table.
- If you don't explicitly associate a subnet with a particular route table, the subnet is implicitly associated with and uses the main route table.
- A subnet can be associated with only one route table at a time, but you can associate multiple subnets with the same route table.
- Use custom route tables for each subnet to permit granular routing for destinations.

# Public Subnet

Public route table

| Destination | Target |
|---|---|
| 172.31.0.0/16 | local |
| 0.0.0.0/0 | Igw-09ef761d872bd7540 |



iNURTURE Education Solutions TOMORROW'S HERE

# Public Subnet

- A public subnet is associated with a route table that has a route to an internet gateway.
- It lets you reach resources inside that subnet from the public internet by assigning public IP addresses.
- Your public subnet configuration acts as a two-way door—allowing traffic to flow either direction, invited or not invited.

# Private Subnet

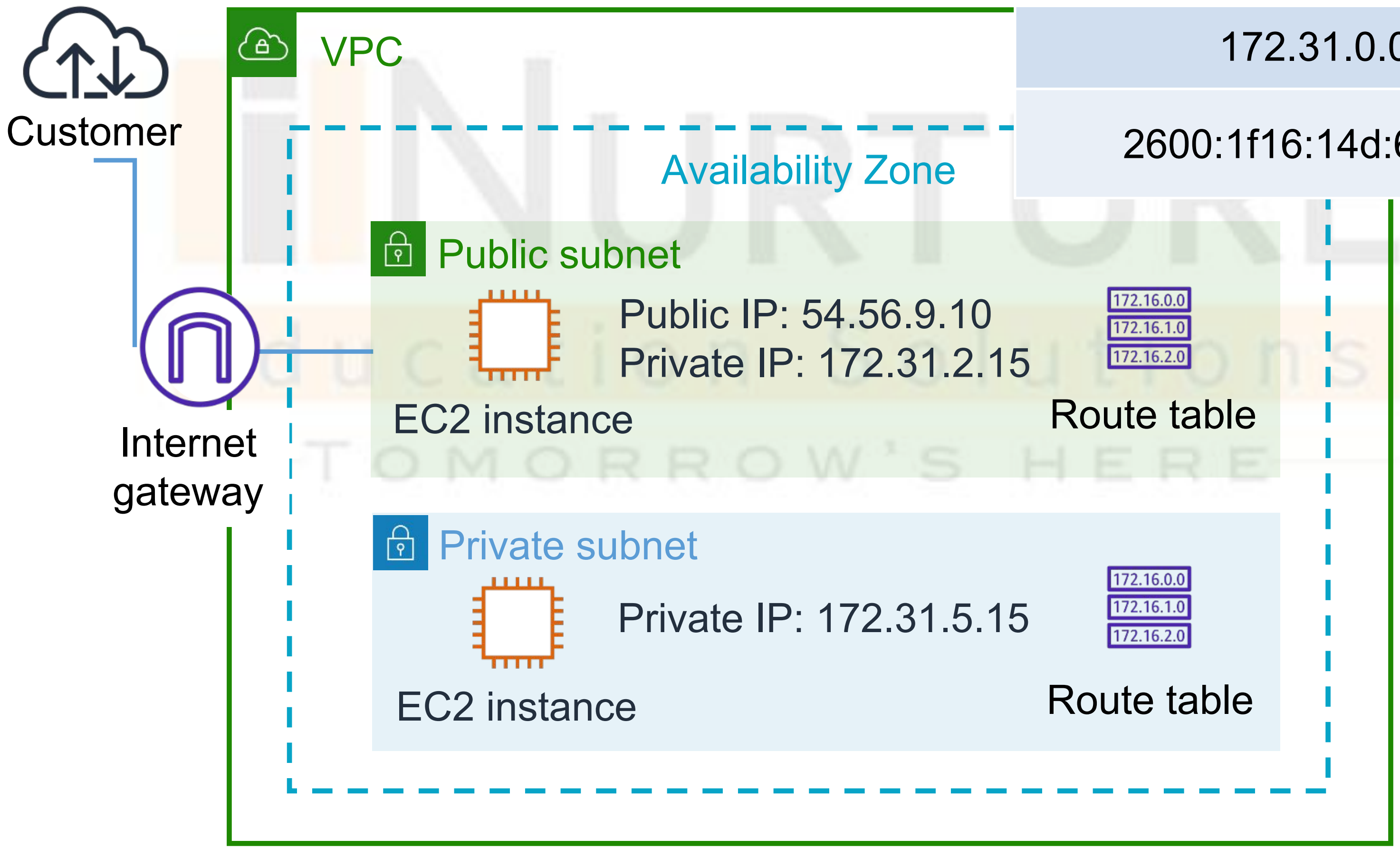
- Private subnets allow indirect access to the internet.
- Traffic stays within your private network.
- A private IP address assigned to an EC2 instance will never change unless you manually assign a new IP address on the network interface of the EC2 instance.
- While you can put web-tier instances into a public subnet, we recommend that you put web-tier instances inside private subnets behind a load balancer placed in a public subnet.
- Some environments require web application instances to be attached to Elastic IP addresses directly.

iNURTURE Education Solutions TOMORROW'S HERE

# Private Subnet

Private route table

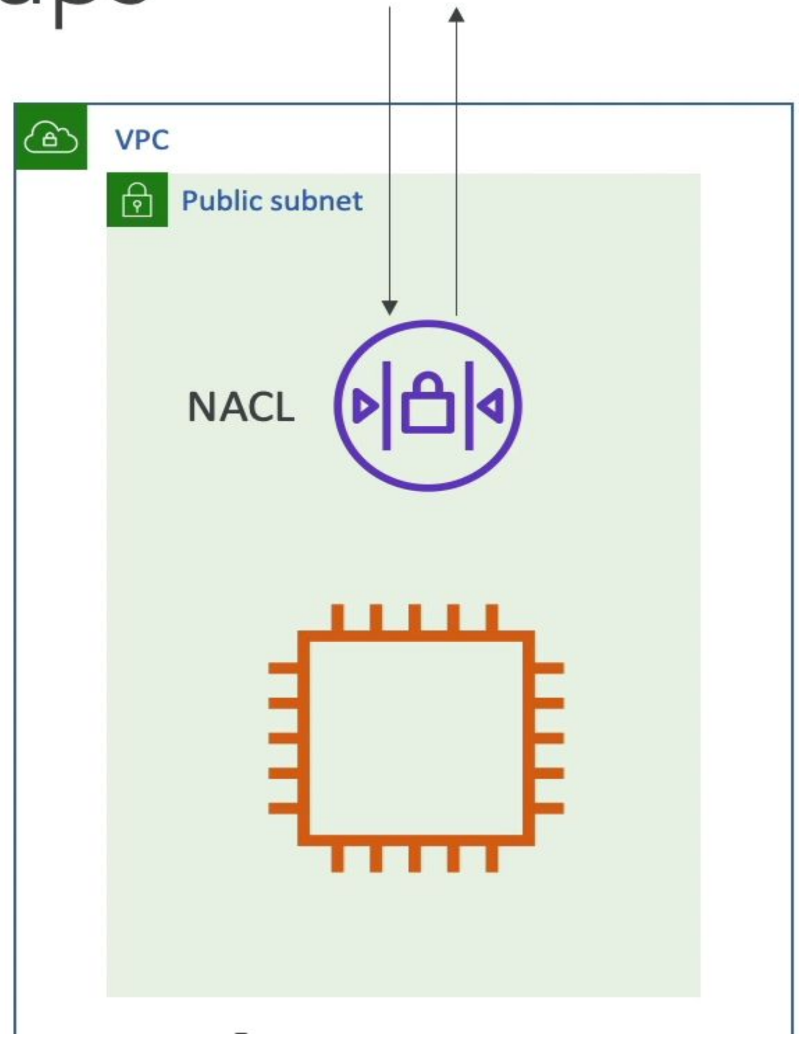
| Destination | Target |
|---|---|
| 172.31.0.0/16 | local |
| 2600:1f16:14d:6300::/56 | local |



iNURTURE Education Solutions TOMORROW'S HERE

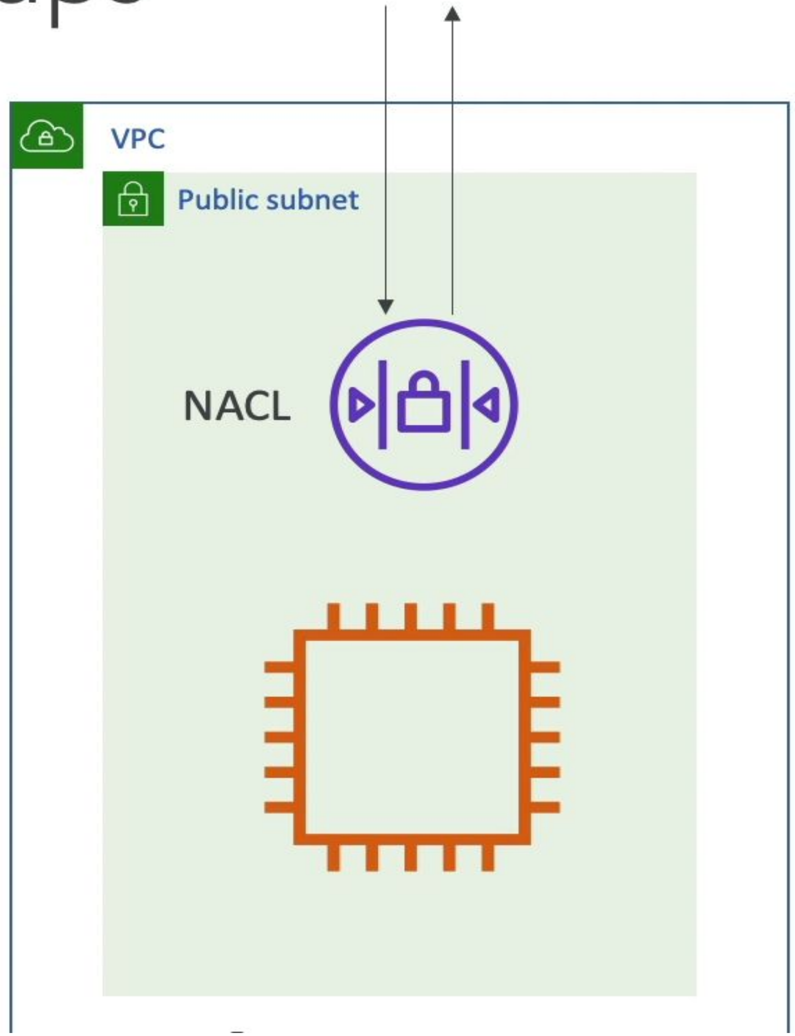
# Network ACL & Security Groups

- NACL (Network ACL)
  - A firewall which controls traffic from and to subnet
  - Can have ALLOW and DENY rules
  - Are attached at the **Subnet** level
  - Rules only include IP addresses

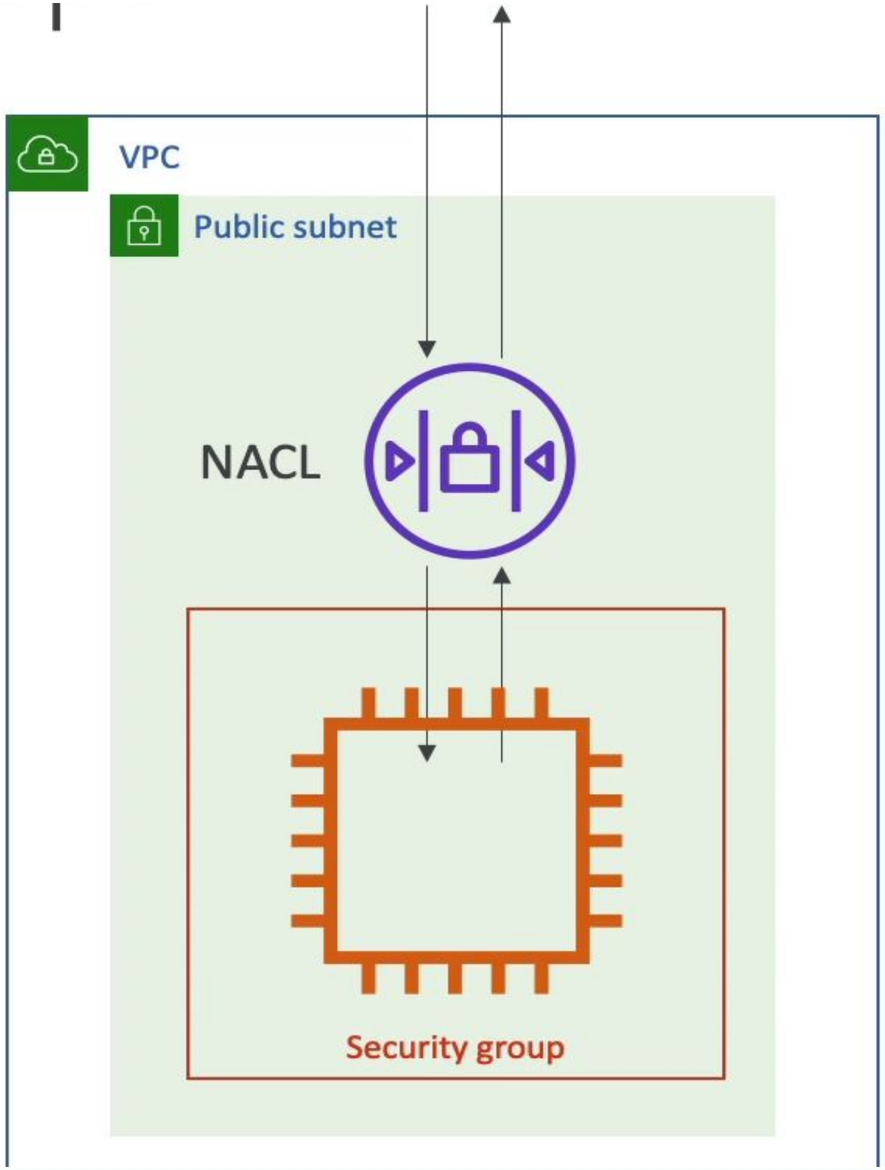


# Network ACL & Security Groups

- NACL (Network ACL)
  - A firewall which controls traffic from and to subnet
  - Can have ALLOW and DENY rules
  - Are attached at the **Subnet** level
  - Rules only include IP addresses



- Security Groups
  - A firewall that controls traffic to and from **an ENI / an EC2 Instance**
  - Can have only ALLOW rules
  - Rules include IP addresses and other security groups

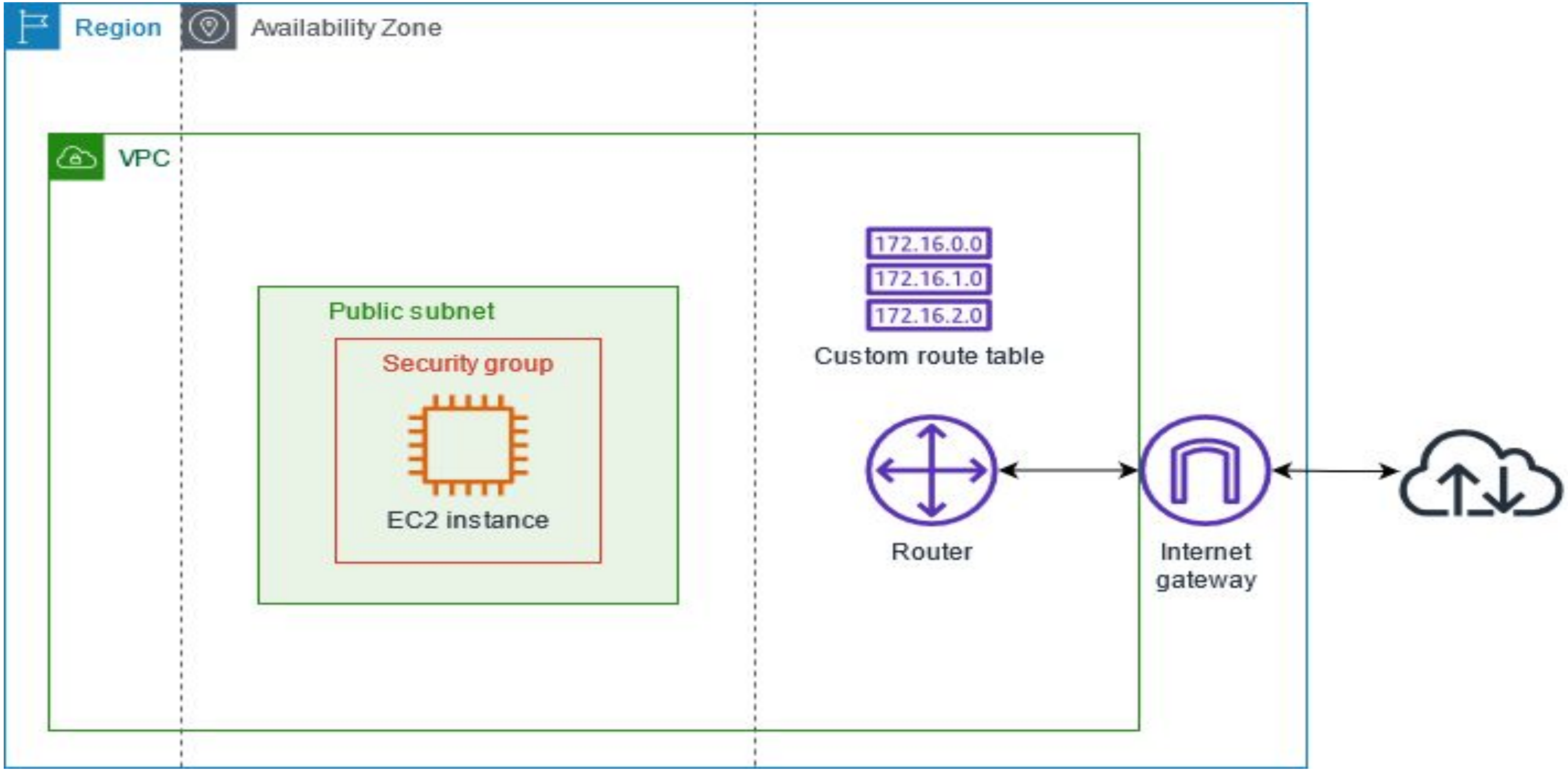


# Scenario 1:VPC With A Public Subnet Only

- The configuration for this scenario includes a virtual private cloud (VPC) with a single public subnet, and an internet gateway to enable communication over the internet.
- Recommend this configuration if you need to run a single-tier, public-facing web application, such as a blog or a simple website.
- Used for Standalone Web

iNURTURE Education Solutions TOMORROW'S HERE

# 1:VPC With A Public Subnet Only



# Scenario 2: VPC with Public And Private Subnets

- The configuration for this scenario includes a virtual private cloud (VPC) with a public subnet and a private subnet.
- Recommend this scenario if you want to run a public-facing web application, while maintaining back-end servers that aren't publicly accessible.
- A common example is a multi-tier website, with the web servers in a public subnet and the database servers in a private subnet.
- You can set up security and routing so that the web servers can communicate with the database servers.
- The instances in the public subnet can send outbound traffic directly to the internet, whereas the instances in the private subnet can't.

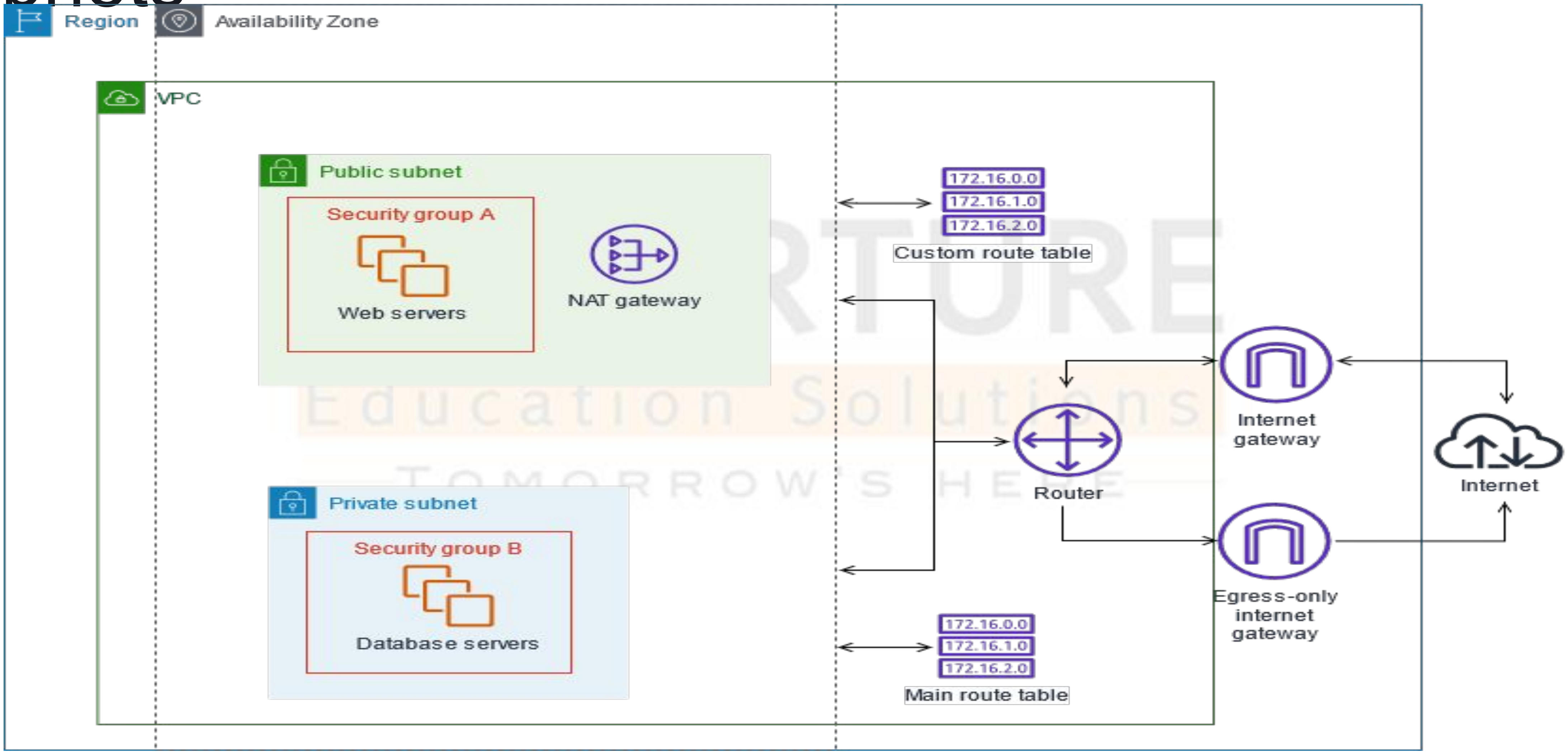
# Scenario 2: VPC with Public And Private Subnets

- The instances in the public subnet can send outbound traffic directly to the internet, whereas the instances in the private subnet can’t.
- Instead, the instances in the private subnet can access the internet by using a network address translation (NAT) gateway that resides in the public subnet.
- The database servers can connect to the internet for software updates using the NAT gateway, but the internet cannot establish connections to the database servers.

# Scenario 2: VPC with Public And Private Subnets

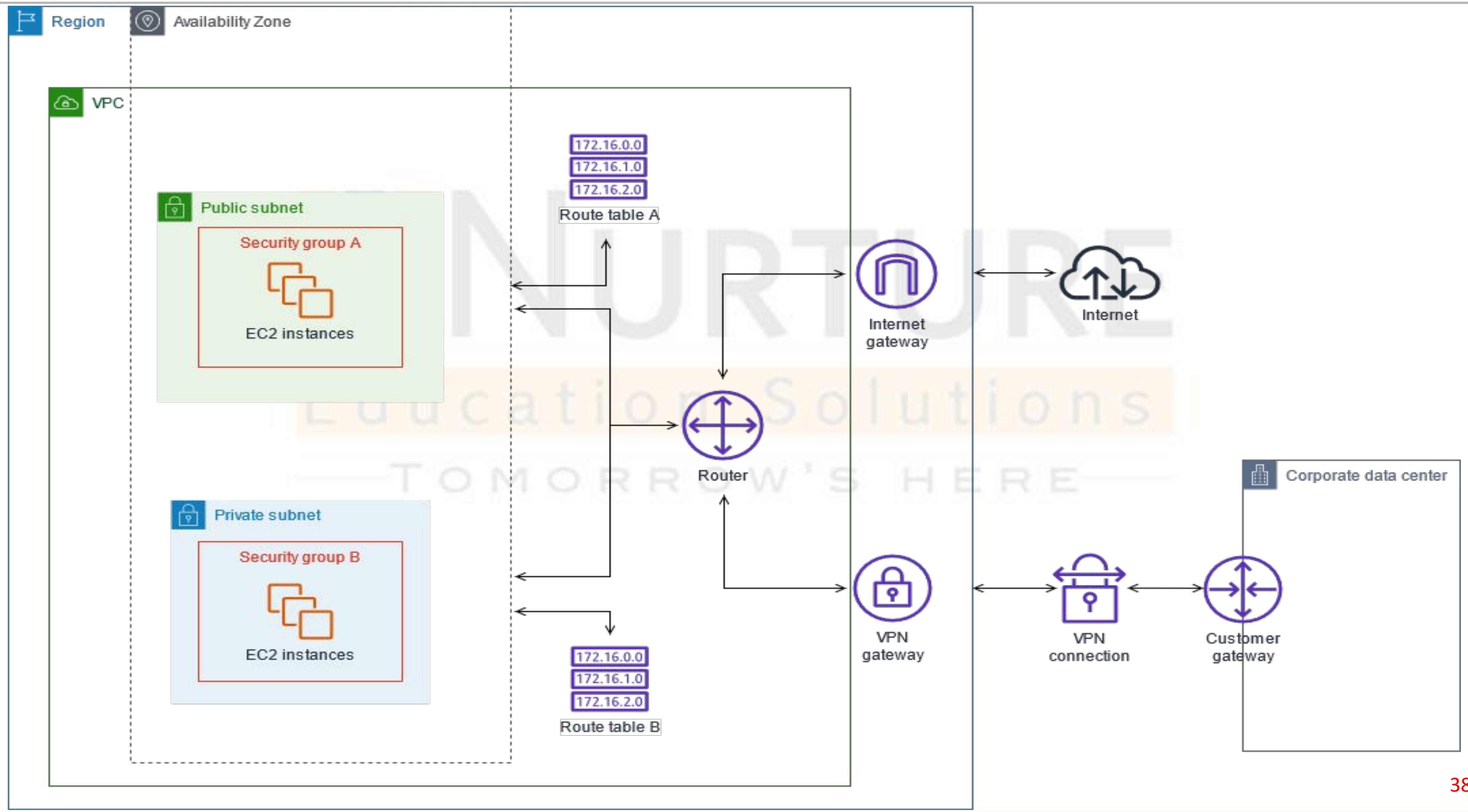
- The instances in the public subnet can send outbound traffic directly to the internet, whereas the instances in the private subnet can’t.
- Instead, the instances in the private subnet can access the internet by using a network address translation (NAT) gateway that resides in the public subnet.
- The database servers can connect to the internet for software updates using the NAT gateway, but the internet cannot establish connections to the database servers.

# Scenario 2: VPC with Public And Private Subnets



# Scenario 3: VPC with public and private subnets and AWS Site-to-Site VPN access

- The configuration for this scenario includes a virtual private cloud (VPC) with a public subnet and a private subnet, and a virtual private gateway to enable communication with your own network over an IPsec VPN tunnel.
- Recommend this scenario if you want to extend your network into the cloud and also directly access the internet from your VPC.
- This scenario enables you to run a multi-tiered application with a scalable web front end in a public subnet, and to house your data in a private subnet that is connected to your network by an IPsec AWS Site-to-Site VPN connection.


Region
Availability Zone
VPC
Public subnet
Security group A
EC2 instances
Private subnet
Security group B
EC2 instances
172.16.0.0
172.16.1.0
172.16.2.0
Route table A
172.16.0.0
172.16.1.0
172.16.2.0
Route table B
Router
Internet gateway
Internet
VPN gateway
VPN connection
Customer gateway
Corporate data center

# The configuration for this scenario includes the following:

- A virtual private cloud (VPC) with a size /16 IPv4 CIDR (example: 10.0.0.0/16). This provides 65,536 private IPv4 addresses.
- A public subnet with a size /24 IPv4 CIDR (example: 10.0.0.0/24). This provides 256 private IPv4 addresses. A public subnet is a subnet that's associated with a route table that has a route to an internet gateway.
- A VPN-only subnet with a size /24 IPv4 CIDR (example: 10.0.1.0/24). This provides 256 private IPv4 addresses.
- An internet gateway. This connects the VPC to the internet and to other AWS products.
- A Site-to-Site VPN connection between your VPC and your network. The Site-to-Site VPN connection consists of a virtual private gateway located on the Amazon side of the Site-to-Site VPN connection and a customer gateway located on your side of the Site-to-Site VPN

iNURTURE Education Solutions TOMORROW'S HERE

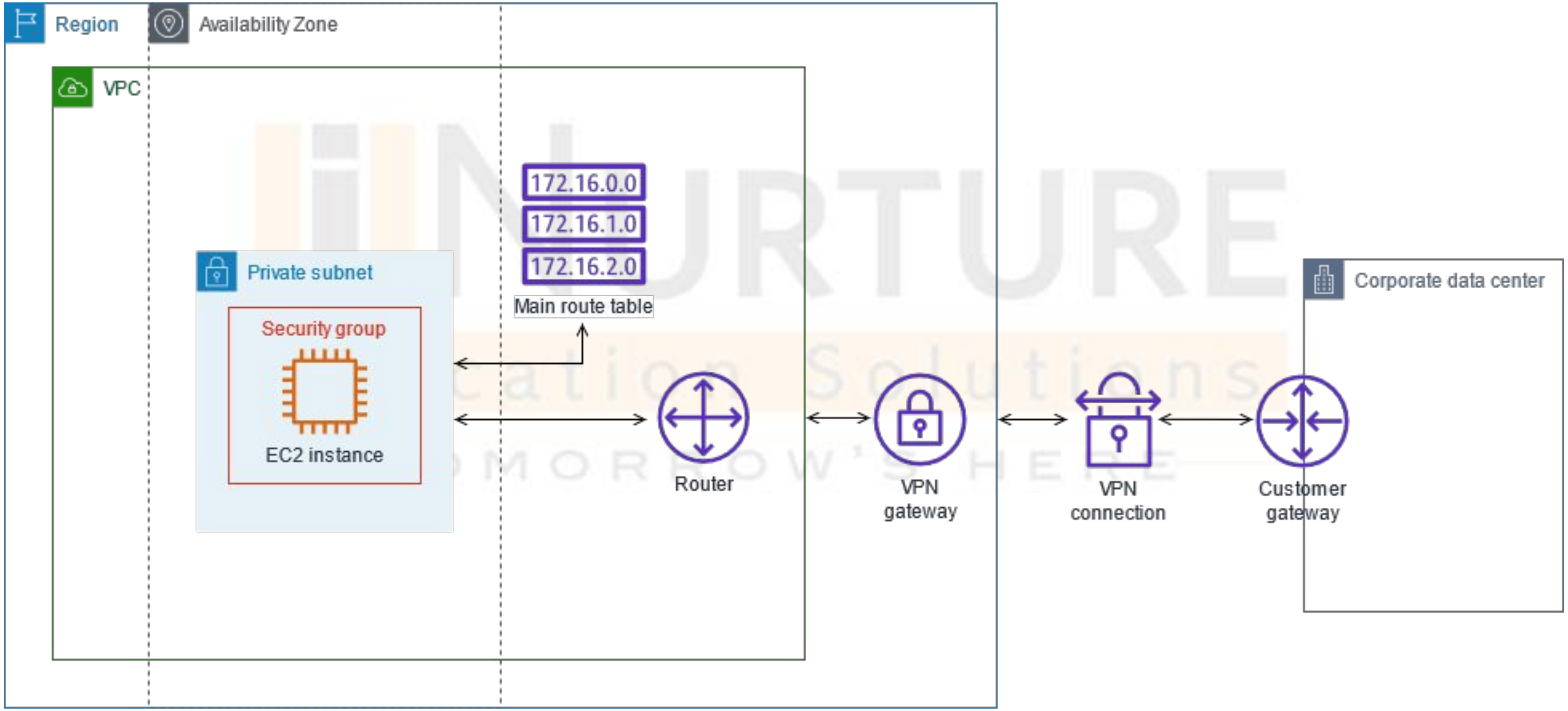
# The configuration for this scenario includes the following:

- Instances with private IPv4 addresses in the subnet range (examples: 10.0.0.5 and 10.0.1.5), which enables the instances to communicate with each other and other instances in the VPC.
- Instances in the public subnet with Elastic IP addresses (example: 198.51.100.1), which are public IPv4 addresses that enable them to be reached from the internet.
- A custom route table associated with the public subnet. This route table contains an entry that enables instances in the subnet to communicate with other instances in the VPC, and an entry that enables instances in the subnet to communicate directly with the internet.
- The main route table associated with the VPN-only subnet. The route table contains an entry that enables instances in the subnet to communicate with other instances in the VPC, and an entry that enables instances in the subnet to communicate directly with your network.

# Scenario 4: VPC With A Private Subnet Only And Hardware VPN Access

- The configuration for this scenario includes a virtual private cloud (VPC) with a single private subnet, and a virtual private gateway to enable communication with your own network over an IPsec VPN tunnel.
- There is no internet gateway to enable communication over the internet.
- Recommend this scenario if you want to extend your network into the cloud using Amazon's infrastructure without exposing your network to the internet.

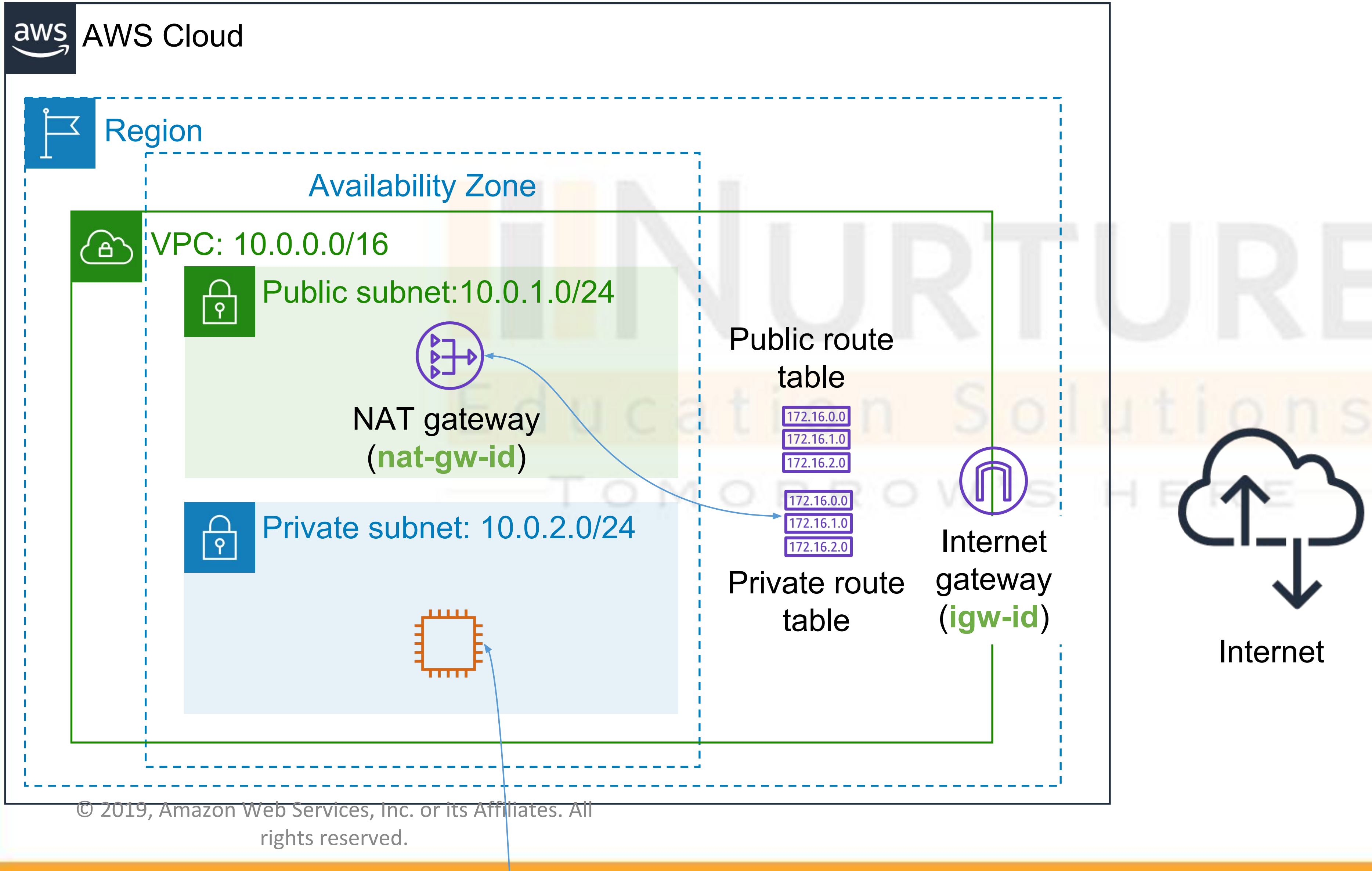
# VPC With A Private Subnet Only And Hardware VPN Access



# The configuration for this scenario includes the following:

- A virtual private cloud (VPC) with a size /16 CIDR (example: 10.0.0.0/16). This provides 65,536 private IP addresses.
- A VPN-only subnet with a size /24 CIDR (example: 10.0.0.0/24). This provides 256 private IP addresses.
- A Site-to-Site VPN connection between your VPC and your network. The Site-to-Site VPN connection consists of a virtual private gateway located on the Amazon side of the Site-to-Site VPN connection and a customer gateway located on your side of the Site-to-Site VPN connection.
- Instances with private IP addresses in the subnet range (examples: 10.0.0.5, 10.0.0.6, and 10.0.0.7), which enables the instances to communicate with each other and other instances in the VPC.
- The main route table contains a route that enables instances in the subnet to communicate with other instances in the VPC.
- Route propagation is enabled, so a route that enables instances in the subnet to communicate directly with your network appears as a propagated route in the main route table.

# Network address translation (NAT) gateway



Public Subnet Route Table

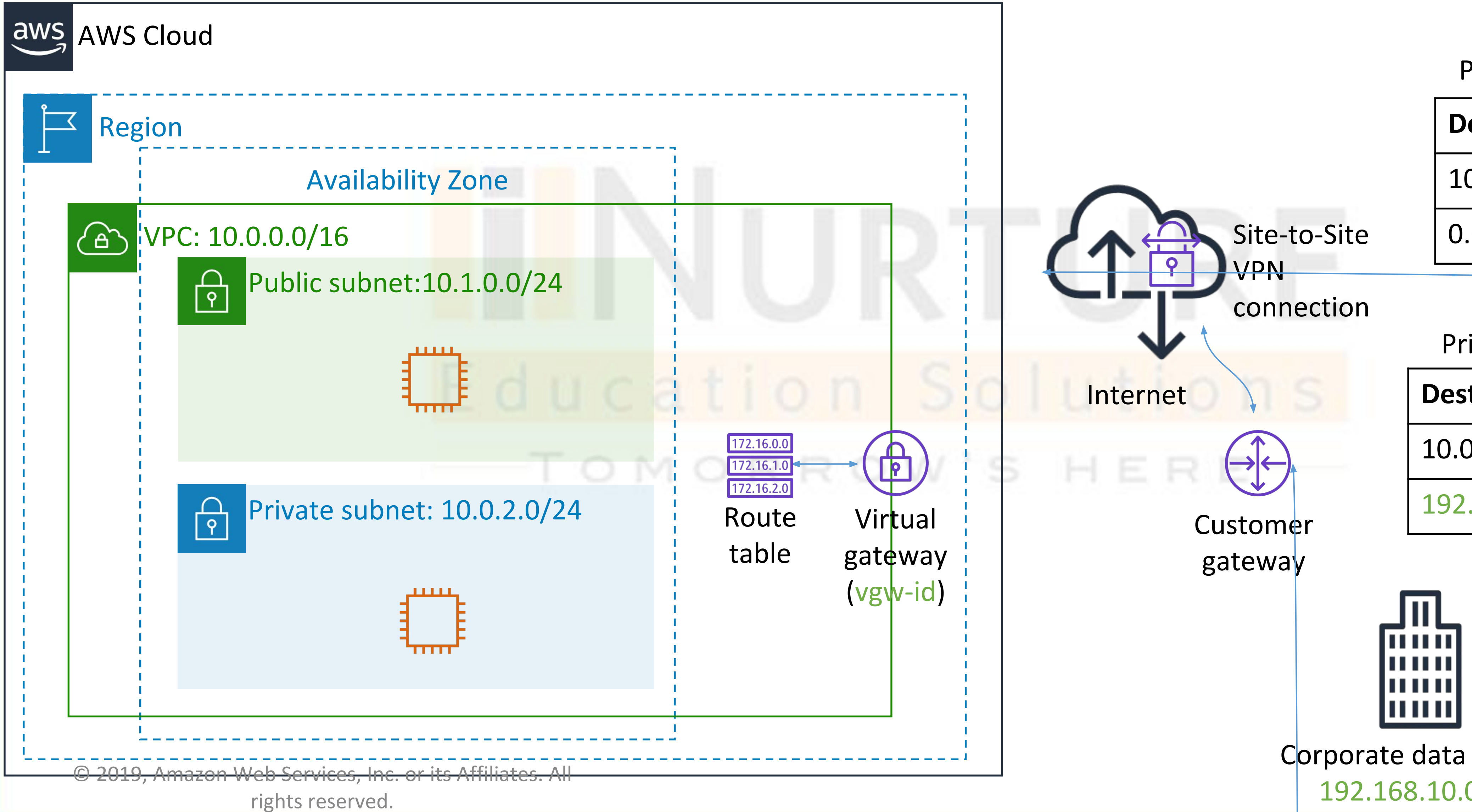
| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 0.0.0.0/0 | igw-id |

Private Subnet Route Table

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

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

# AWS Site-to-Site VPN



Public subnet route table

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

Private subnet route table

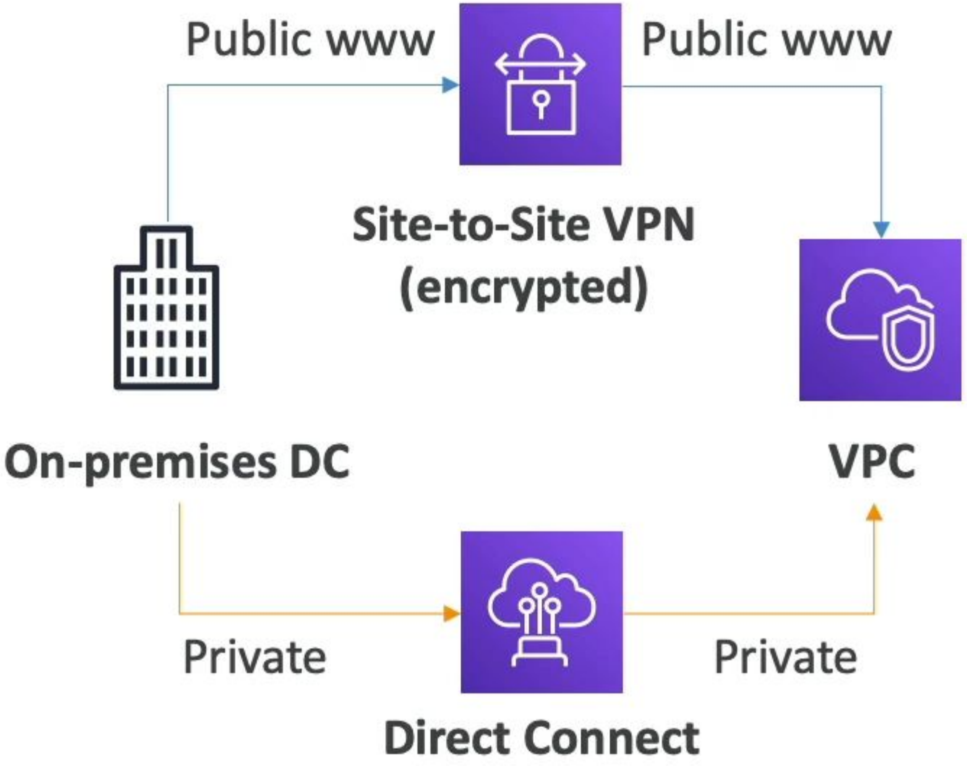
| Destination | Target |
|---|---|
| 10.0.0.0/16 | local |
| 192.168.10.0/24 | vgw-id |

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

# AWS Site-to-Site VPN

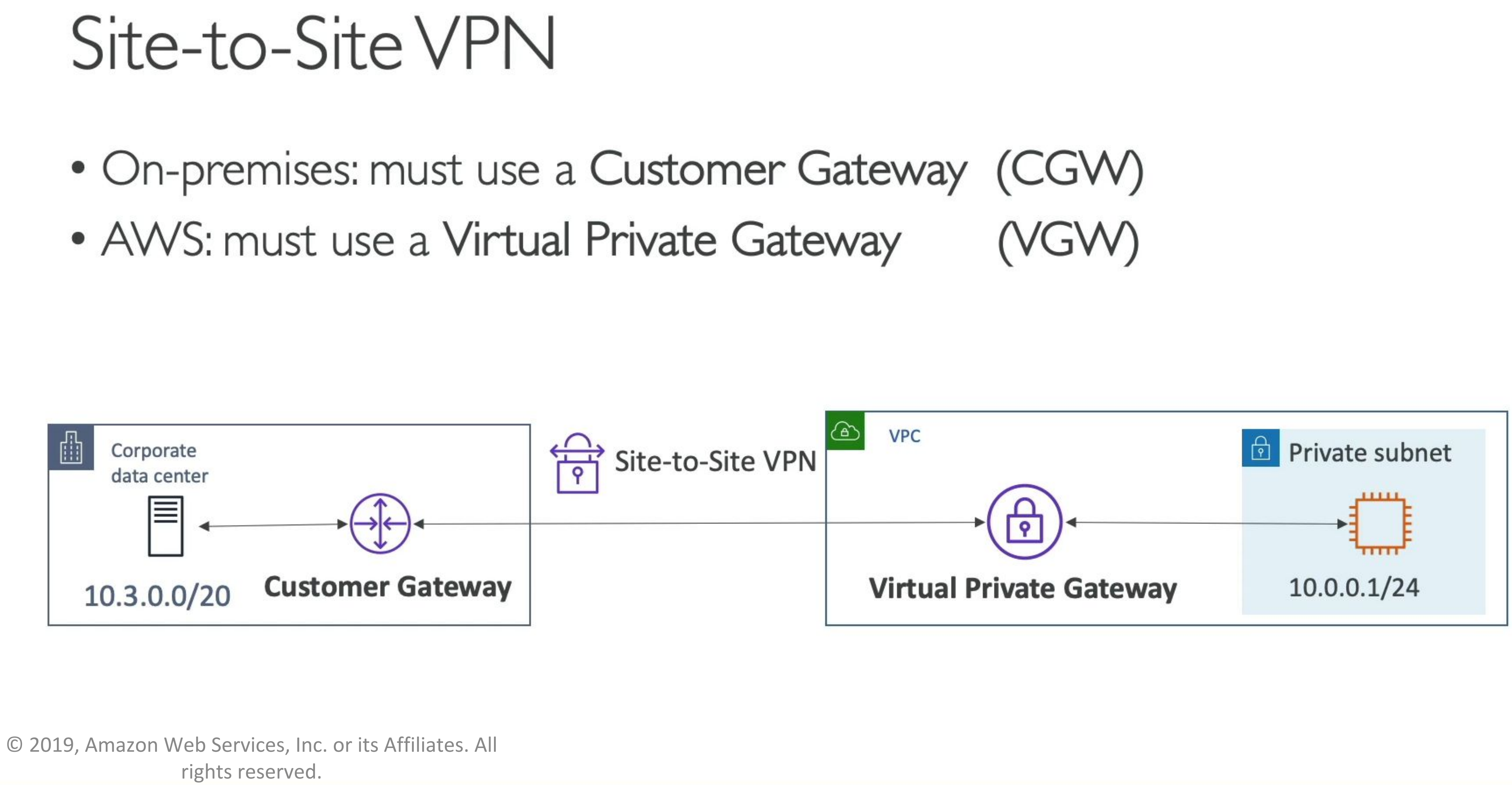
## Site to Site VPN & Direct Connect

- Site to Site VPN
  - Connect an on-premises VPN to AWS
  - The connection is automatically encrypted
  - Goes over the public internet
- Direct Connect (DX)
  - Establish a physical connection between on-premises and AWS
  - The connection is private, secure and fast
  - Goes over a private network
  - Takes at least a month to establish



rights reserved.

# AWS Site-to-Site VPN



## Site-to-Site VPN

- On-premises: must use a Customer Gateway (CGW)
- AWS: must use a Virtual Private Gateway (VGW)

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

# The configuration for this scenario includes the following:

- By default, instances that you launch into a VPC cannot communicate with a remote network. To connect your VPC to your remote network (that is, create a virtual private network or VPN connection), you:

1. Create a new virtual gateway device (called a *virtual private network (VPN) gateway*) and attach it to your VPC.
2. Define the configuration of the VPN device or the *customer gateway*. The customer gateway is not a device but an AWS resource that provides information to AWS about your VPN device.
3. Create a custom route table to point corporate data center-bound traffic to the VPN gateway. You also must update security group rules. (You will learn about security groups in the next section.)
4. Establish an *AWS Site-to-Site VPN (Site-to-Site VPN) connection* to link the two systems together.
5. Configure routing to pass traffic through the connection.

# Amazon Route 53

- Amazon Route 53 is a highly available and scalable Domain Name System (DNS) web service. Route 53 connects user requests to internet applications running on AWS or on-premises.
- Amazon Route 53, part of the Amazon Web Services (AWS) cloud computing platform from Amazon.com normally referred to as AWS Route 53, is a highly available, scalable Domain Name System (DNS) service.
- Released in 2010, its name refers to both the classic highway US Route 66 and the destination for DNS server requests: TCP or UDP port 53.
- AWS Route 53 translates URL names, such as www.wordpress.com, into their corresponding numeric IP addresses—in this example, 198.143.164.252. In this way, AWS Route 53 simplifies how cloud architecture routes users to internet applications.

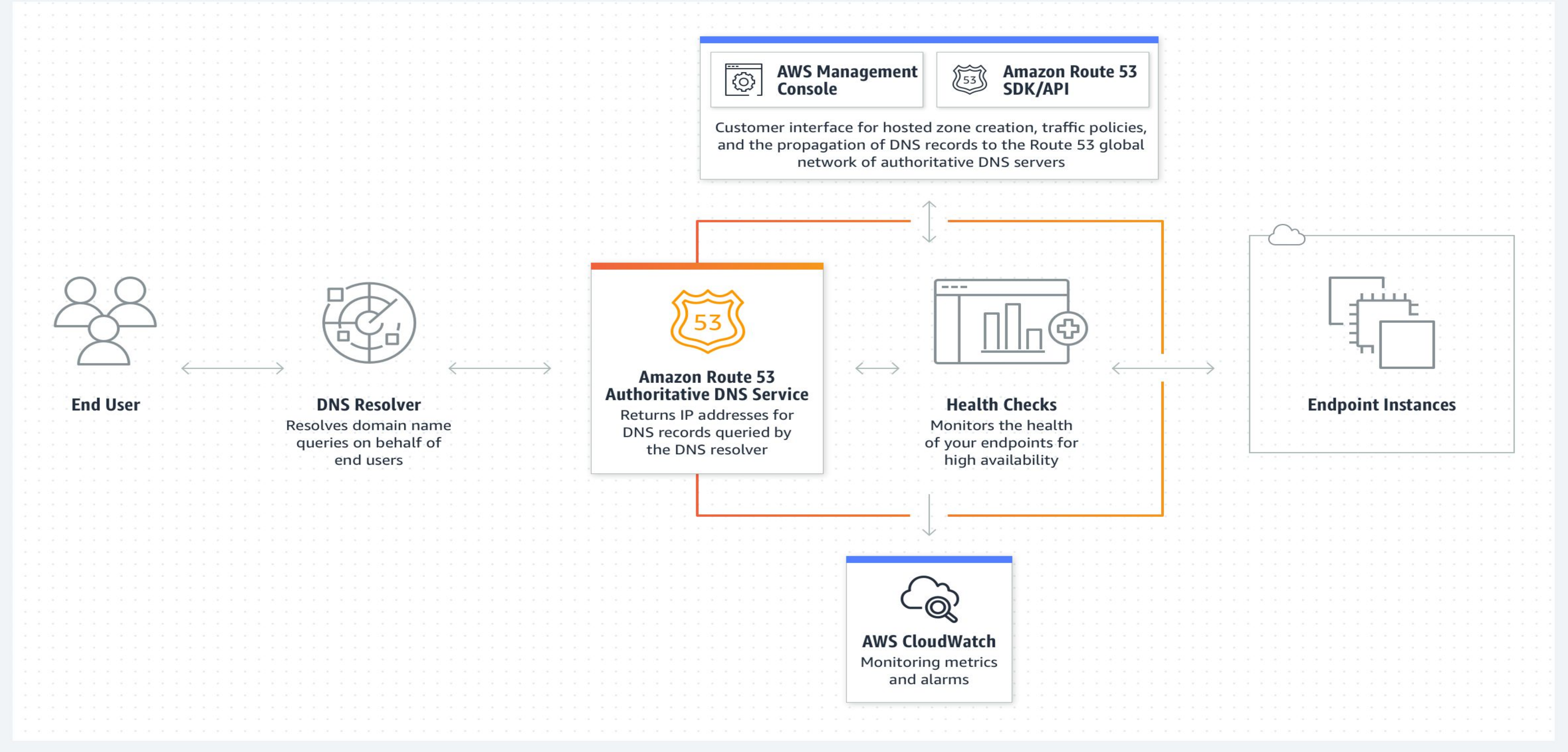
# Amazon Route 53

- **You can use Route 53 to register new domain, transfer existing domain, route traffic for your domain to AWS and external resource and monitor the health of your resource.**
- The AWS Route 53 DNS service connects user requests to ELB load balancers, Amazon EC2 instances, Amazon S3 buckets, and other infrastructure running on AWS.

## Route 53 Functions

1. **DNS Management**
2. **Traffic Management**
3. **Availability Monitoring**
4. **Domain registration**

# Amazon Route 53



AWS Management Console

Amazon Route 53 SDK/API

Customer interface for hosted zone creation, traffic policies, and the propagation of DNS records to the Route 53 global network of authoritative DNS servers

**End User**

**DNS Resolver**

Resolves domain name queries on behalf of end users

**Amazon Route 53 Authoritative DNS Service**

Returns IP addresses for DNS records queried by the DNS resolver

**Health Checks**

Monitors the health of your endpoints for high availability

**Endpoint Instances**

**AWS CloudWatch**

Monitoring metrics and alarms

# Amazon Route 53

The following process occurs when a user accesses a web server via Route 53 DNS:

- A user accesses an address managed by Route 53, www.website.com, which leads to an AWS-hosted machine.
- Typically managed by the local network or ISP, the user's DNS resolver receives the request for www.website.com routed by AWS Route 53 and forwards it to a DNS root server.
- The DNS resolver forwards the TLD name servers for ".com" domains the user requests.
- The resolver acquires the four authoritative Amazon Route 53 name servers that host the domain's DNS zone.
- The DNS resolver selects one of the four AWS Route 53 servers, and requests details for www.website.com.
- The Route 53 name server searches the DNS zone for the www.website.com IP address and other relevant information and returns it to the DNS resolver.
- As specified by the Time to Live (TTL) parameter, the DNS resolver caches the IP address locally, and of course returns it to the user's web browser.
- The browser uses the IP address the resolver provides to contact Amazon-hosted services such as the web server.
- The user's web browser displays the website.

# Amazon Route 53

Amazon Route 53 Benefits and Features

- AWS service integration: the tight integration of AWS Route 53 with CloudFront, S3, and ELB means it's easy to route traffic to a static website hosted on S3 or an ELB CNAME record, or generate custom domains for CloudFront URLs.
- Simple routing policy: The simplest and most common routing type, this policy merely uses AWS Route 53 to map your site name to your IP. Any future browser requests for that site name would then be directed to the correct IP.
- Amazon Route 53 failover: In case of outage as determined by health checks, an Amazon Route 53 failover policy redirects users to a designated backup resource or alternative service automatically.

# Amazon Route 53

iNURTURE Education Solutions TOMORROW'S HERE

Amazon Route 53 Benefits and Features

- Domain registration: AWS serves as a domain registrar, allowing users to select and register domain names from all top-level domains (.com, .net, .org, etc.) with the AWS management console.
- Health checks: AWS Route 53 conducts health checks and monitors the health and performance of applications. When it detects an outage, Amazon Route 53 redirects users to a healthy resource.
- Latency-based routing: A latency-based policy routes users and traffic to the lowest latency AWS region.

# Amazon Route 53 Overview

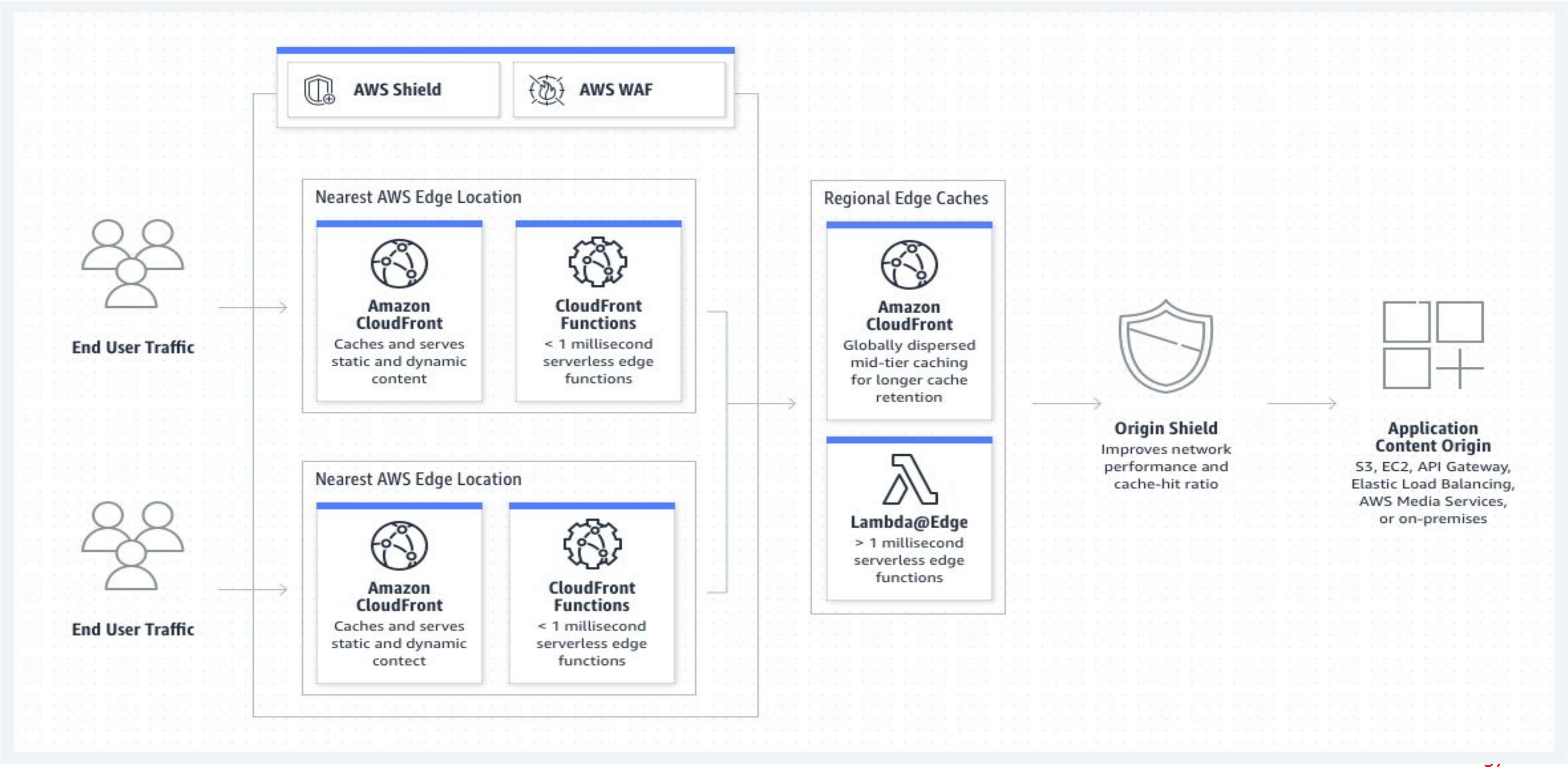
- Route53 is a Managed DNS (Domain Name System)
- DNS is a collection of rules and records which helps clients understand how to reach a server through URLs.

- In AWS, the most common records are:
  - www.google.com => 12.34.56.78 == A record (IPv4)
  - www.google.com => 2001:0db8:85a3:0000:0000:8a2e:0370:7334 == AAAA IPv6
  - search.google.com => www.google.com == CNAME: hostname to hostname
  - example.com => AWS resource == Alias (ex: ELB, CloudFront, S3, RDS, etc…)

# CloudFront

- Amazon CloudFront is a webserver that gives business and web application developers an easy and cost effective way to distribute content with low latency and high data transfer speed.
- Securely deliver content with low latency and high transfer speeds
- Amazon CloudFront is a content delivery network (CDN) service built for high performance, security, and developer convenience.

# CloudFront



AWS Shield

AWS WAF

End User Traffic

Nearest AWS Edge Location

**Amazon CloudFront**
Caches and serves static and dynamic content

**CloudFront Functions**
< 1 millisecond serverless edge functions

End User Traffic

Nearest AWS Edge Location

**Amazon CloudFront**
Caches and serves static and dynamic contect

**CloudFront Functions**
< 1 millisecond serverless edge functions

Regional Edge Caches

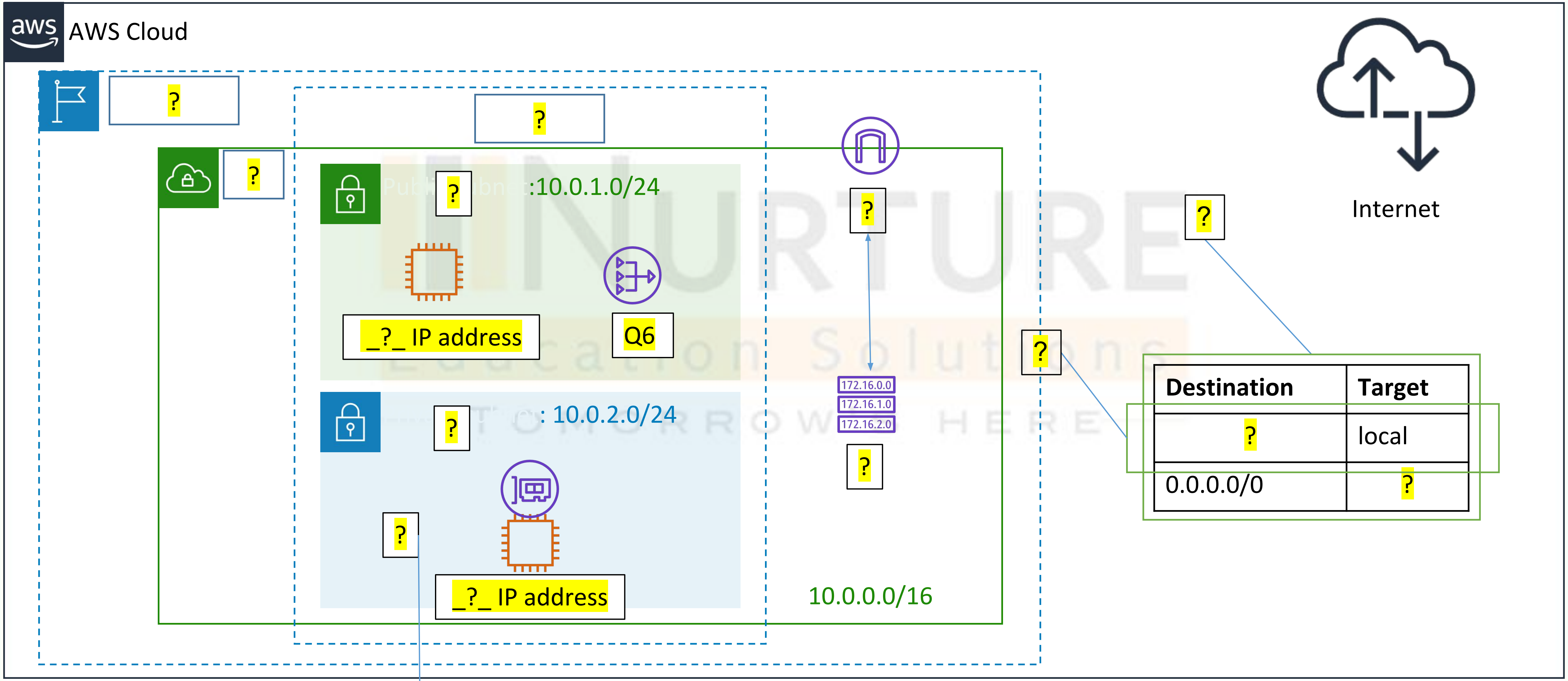
**Amazon CloudFront**
Globally dispersed mid-tier caching for longer cache retention

**Lambda@Edge**
> 1 millisecond serverless edge functions

**Origin Shield**
Improves network performance and cache-hit ratio

**Application Content Origin**
S3, EC2, API Gateway, Elastic Load Balancing, AWS Media Services, or on-premises

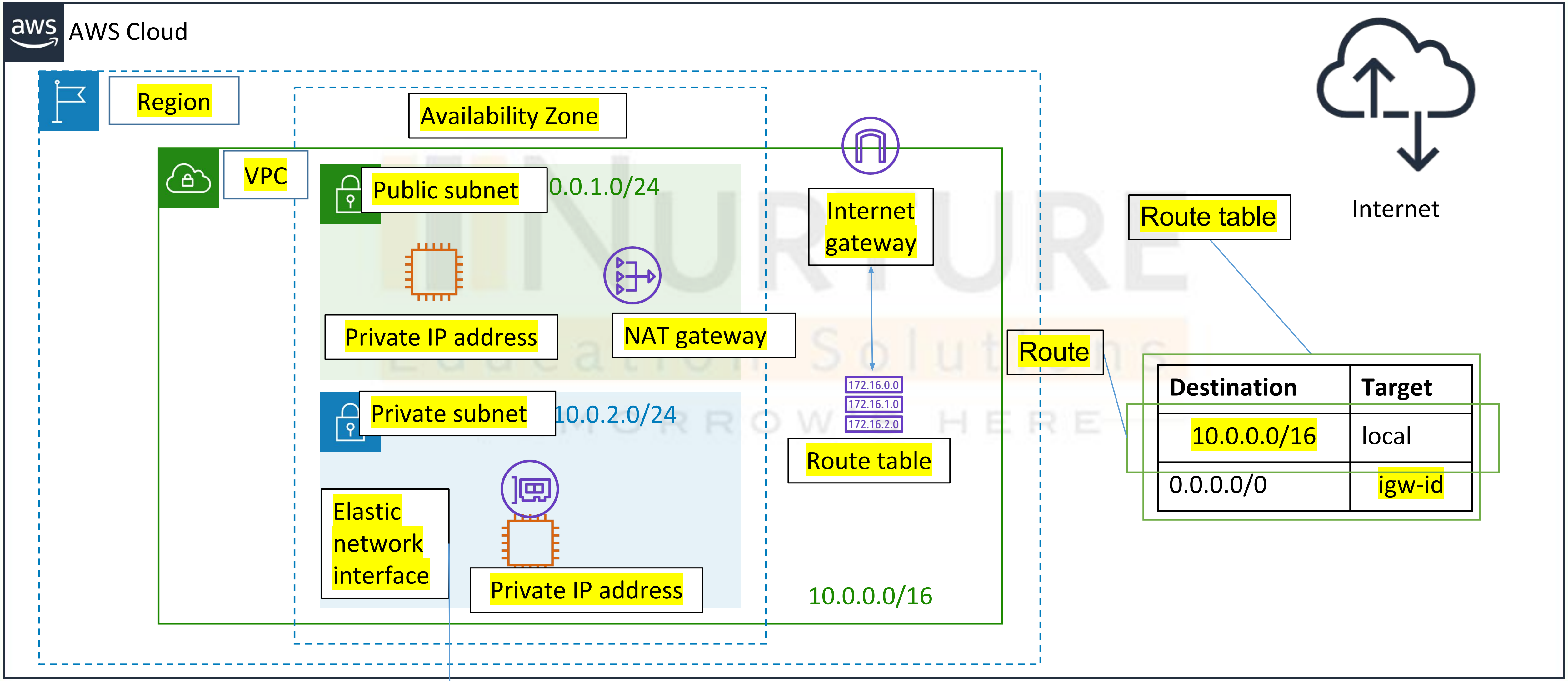
# Activity: Label this network diagram



| Destination | Target |
|---|---|
| ? | local |
| 0.0.0.0/0 | ? |

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

# Activity: Solution



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

© 2019, Amazon Web Services, Inc. or its Affiliates. All rights reserved.

# References

- https://network00.com/NetworkTools/IPv4AddressPlanner/
- AWS Academy