



COMP5349– Cloud Computing

Week 5: Cloud Networking

Dr. Ying Zhou

The University of Sydney

Table of Contents

COMMONWEALTH OF AUSTRALIA

Copyright Regulations 1969

WARNING

This material has been reproduced and communicated to you by or on behalf of the **University of Sydney** pursuant to Part VB of the Copyright Act 1968 (the Act).

The material in this communication may be subject to copyright under the Act. Any further reproduction or communication of this material by you may be the subject of copyright protection under the Act.

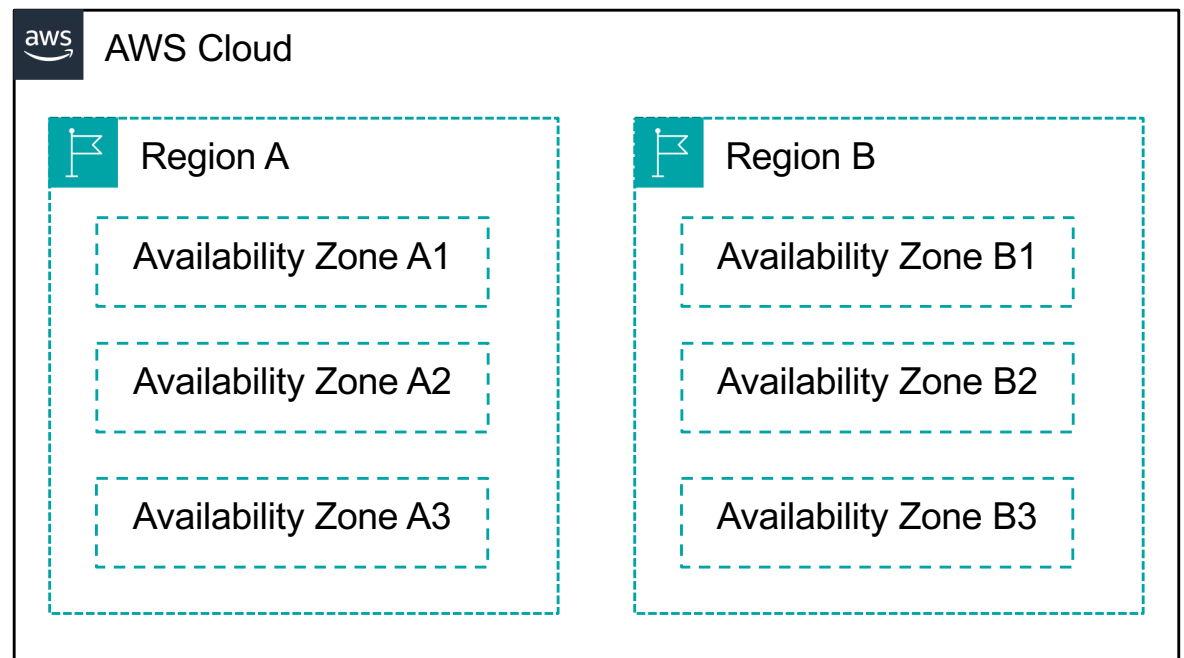
Do not remove this notice

- 01** Amazon VPC Basic Concepts
- 02** VPC Routing
- 03** VPC Security
- 04** Connecting to Managed Services
- 05** AWS Internal traffic
- 06** AWS external traffic

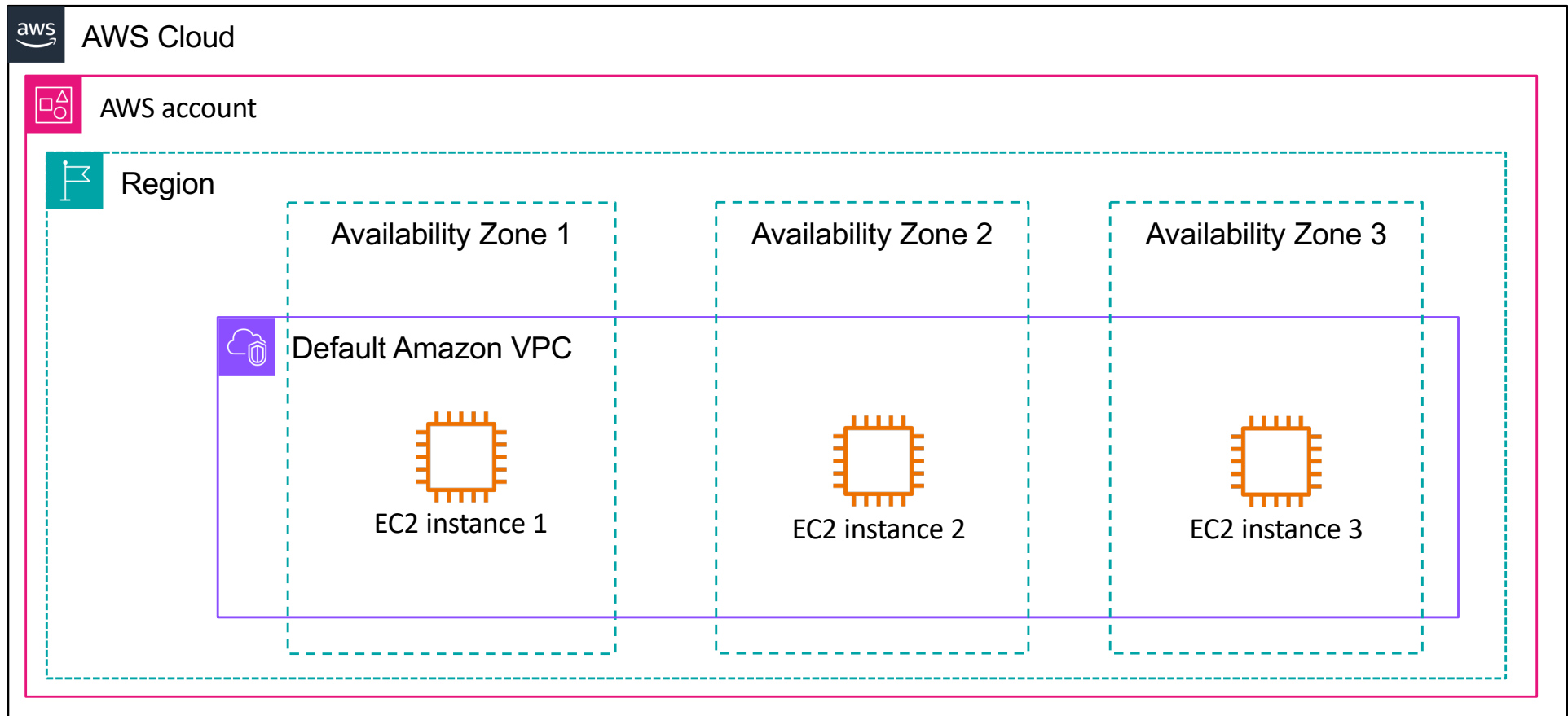
Amazon VPC Basic Concepts

AWS physical infrastructure

- AWS Cloud infrastructure resides in data centers which contain thousands of servers built into racks. Every rack has network routers and switches to route traffic.
- Data centers are grouped together in Availability Zones (AZs).
- AZs are connected with single digit millisecond latency network.
- AZs are grouped together in an AWS Region.
- Latency between AWS Regions is 10s of milliseconds.

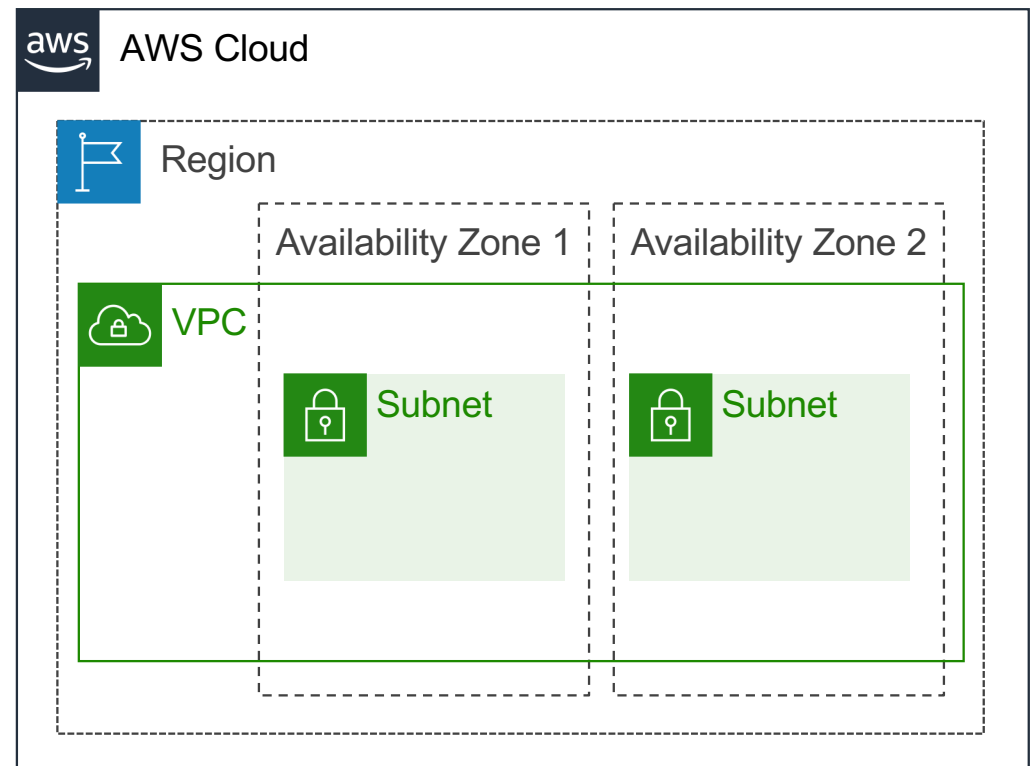


Review: AWS account resource isolation




Review: VPCs and subnets

- VPCs:
 - **Logically isolated** from other VPCs
 - **Dedicated** to your AWS account
 - Belong to a single **AWS Region** and can span multiple Availability Zones
- Subnets:
 - **Range of IP addresses** that divide a VPC
 - Belong to a single **Availability Zone**
 - Classified as **public** or **private**



IP addressing

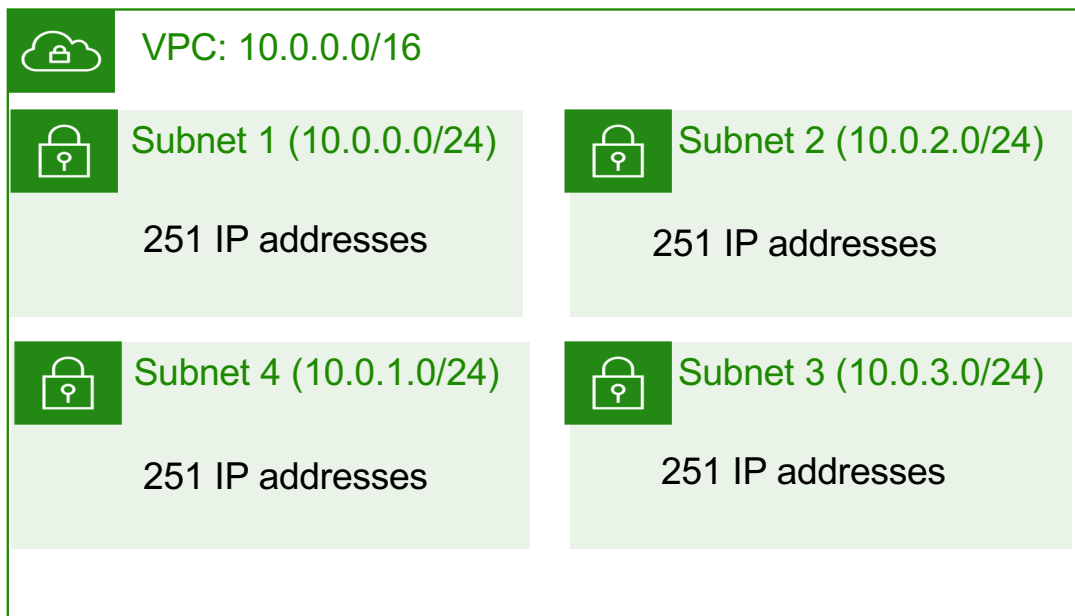
- When you create a VPC, you assign it to an IPv4 **CIDR block** (range of **private** IPv4 addresses).
- You **cannot change the address range** after you create the VPC.
- The **largest** IPv4 CIDR block size is **/16**.
- The **smallest** IPv4 CIDR block size is **/28**.
- IPv6 is also supported (with a different block size limit).
- CIDR blocks of subnets **cannot overlap**.

 VPC

x.x.x.x/16 or 65,536 addresses (max)
to
x.x.x.x/28 or 16 addresses (min)

Reserved IP addresses

Example: A VPC with an IPv4 CIDR block of 10.0.0.0/16 has 65,536 total IP addresses. The VPC has four equal-sized subnets. Only 251 IP addresses are available for use by each subnet.



IP Addresses for CIDR block 10.0.0.0/24	Reserved for
10.0.0.0	Network address
10.0.0.1	Internal communication
10.0.0.2	Domain Name System (DNS) resolution
10.0.0.3	Future use
10.0.0.255	Network broadcast address

Public IP address types

Public IPv4 address

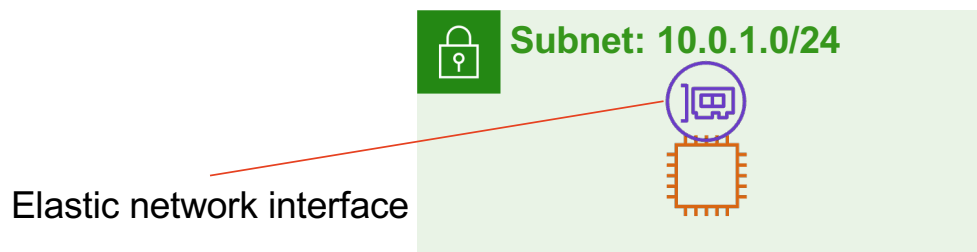
- Manually assigned through an Elastic IP address
- Automatically assigned through the auto-assign public IP address settings at the subnet level

Elastic IP address

- Associated with an AWS account
- Can be allocated and remapped anytime
- Additional costs might apply

Elastic network interface

- An elastic network interface is a **virtual network interface** that you can:
 - Attach to an instance.
 - Detach from the instance, and attach to another instance to redirect network traffic.
- Its **attributes follow** when it is reattached to a new instance.
- Each instance in your VPC has a **default network interface** that is assigned a private IPv4 address from the IPv4 address range of your VPC.
- An instance can have multiple ENIs
- The ENI's network traffic is processed by the Nitro Card for VPC on the physical host



VPC Routing

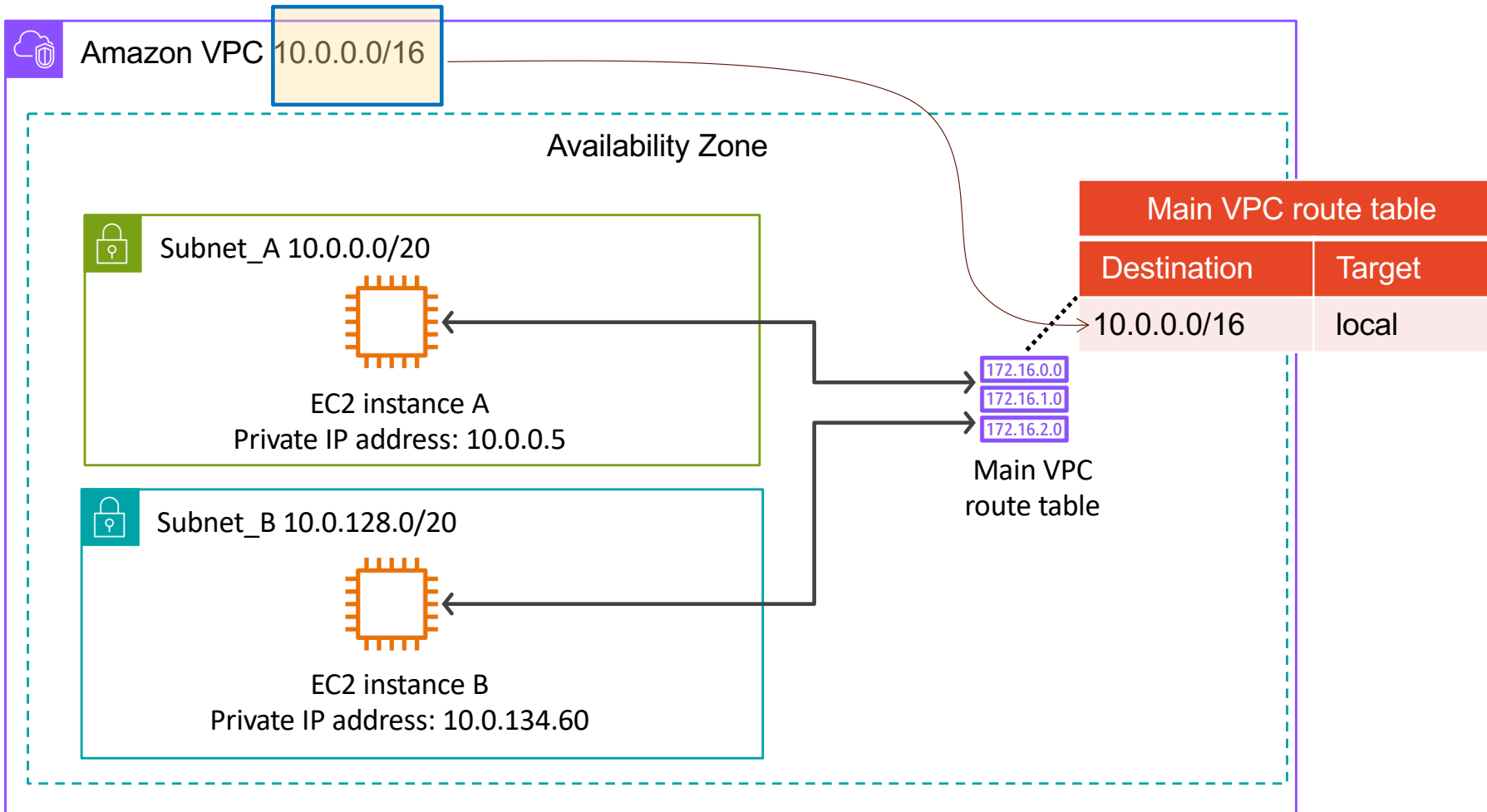
Route tables and route

- **Route tables are essential for controlling network traffic flow.** They contain a set of rules, or routes, that determine where network traffic is directed.
- Each **route** specifies a *destination* and a *target*.
 - **Destination:** The CIDR block (IP address range) specifying *where* the traffic is going.
 - **Target:** The *next hop* or device that the traffic should be sent to in order to reach the destination.
- **Analogy: Home Network Example**
 - **Scenario:** Accessing a website like google.com.
 - **Destination:** The IP address(es) associated with google.com.
 - **Target:** Your home router.
 - **Explanation:** All traffic from devices within your home network destined for the internet is first routed through your router.

The Main Route Table in an AWS VPC

- **Default Traffic Control:** Every VPC automatically comes with a "main route table."
 - This route table acts as the default traffic director for all subnets within the VPC, unless a subnet is explicitly associated with a custom route table.
- **Initial Configuration:** By default, the main route table contains a local route, allowing communication within the VPC itself.
 - This local route typically has a destination of the VPC's CIDR block and a target of "local".
 - The "local" target is used for routes that allow communication between subnets within the same VPC.

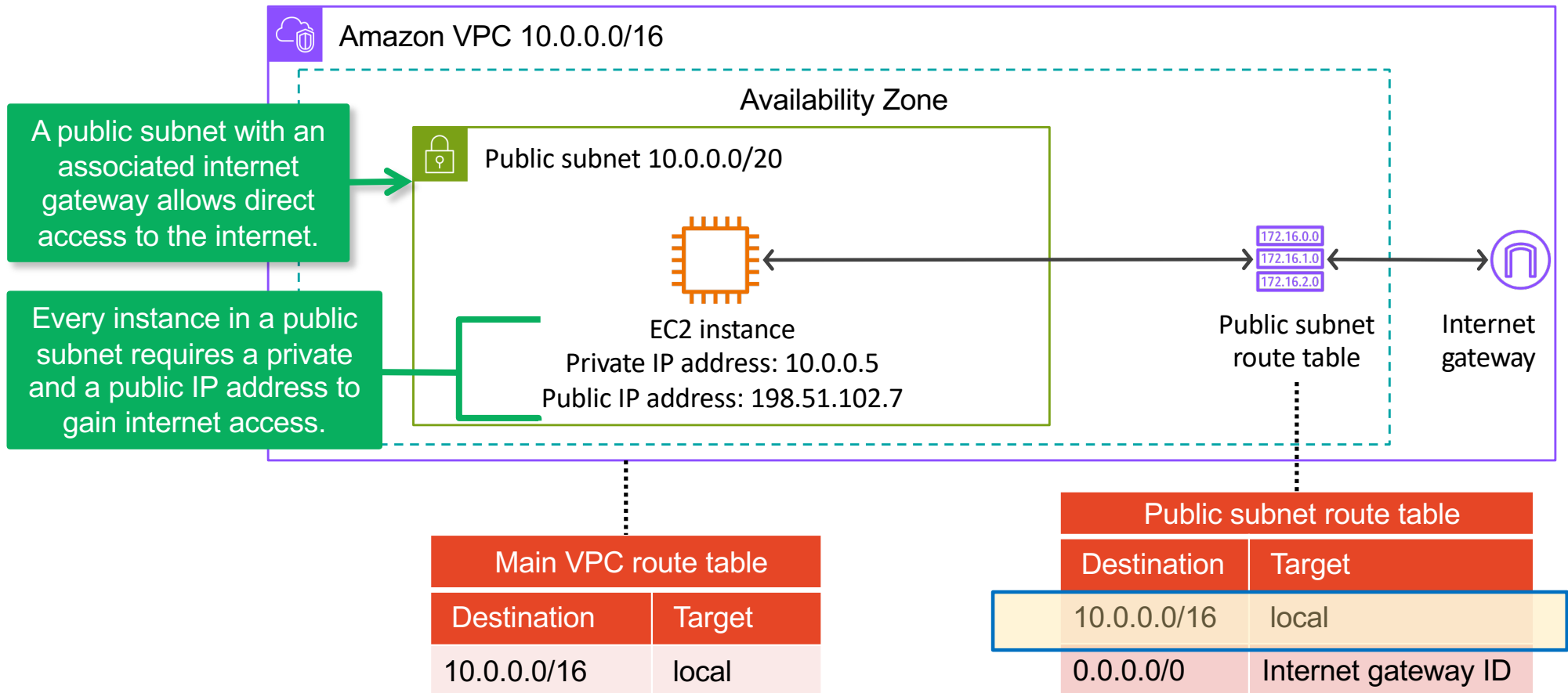
Main route table



Subnet and Route table

- **Subnet Association:** When you create a new subnet in your VPC, if you don't specify a route table, it's automatically associated with the main route table.
 - This is the reason that all subnets within a VPC can communicate with each other by default.
- **Customization:** You can modify the main route table to add routes for internet access, or other destinations.
 - However, it is considered best practice to create custom route tables, and leave the main route table as the default.
- Each subnet within a VPC can only be associated with one route table at a time.

Public subnets

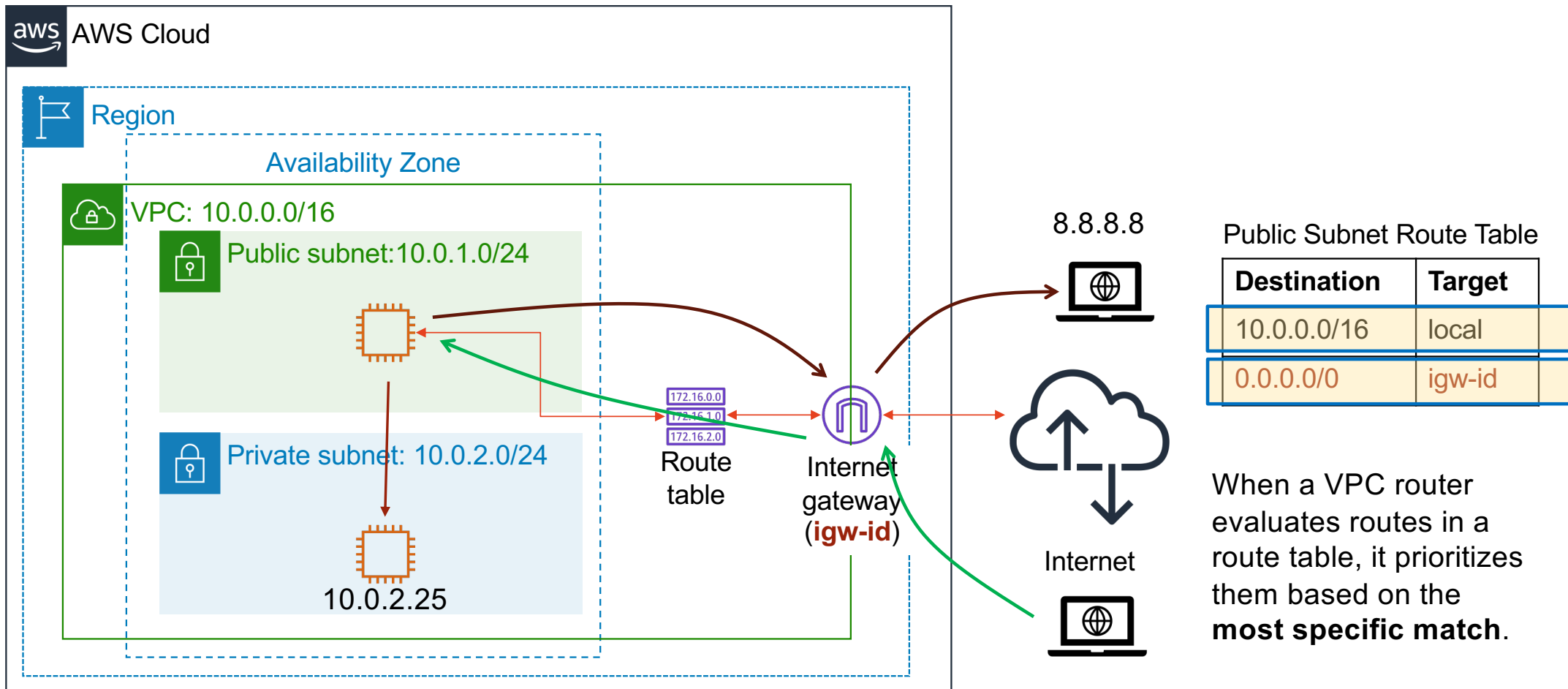


The local route is still needed because this is the only route table associated with the public subnet

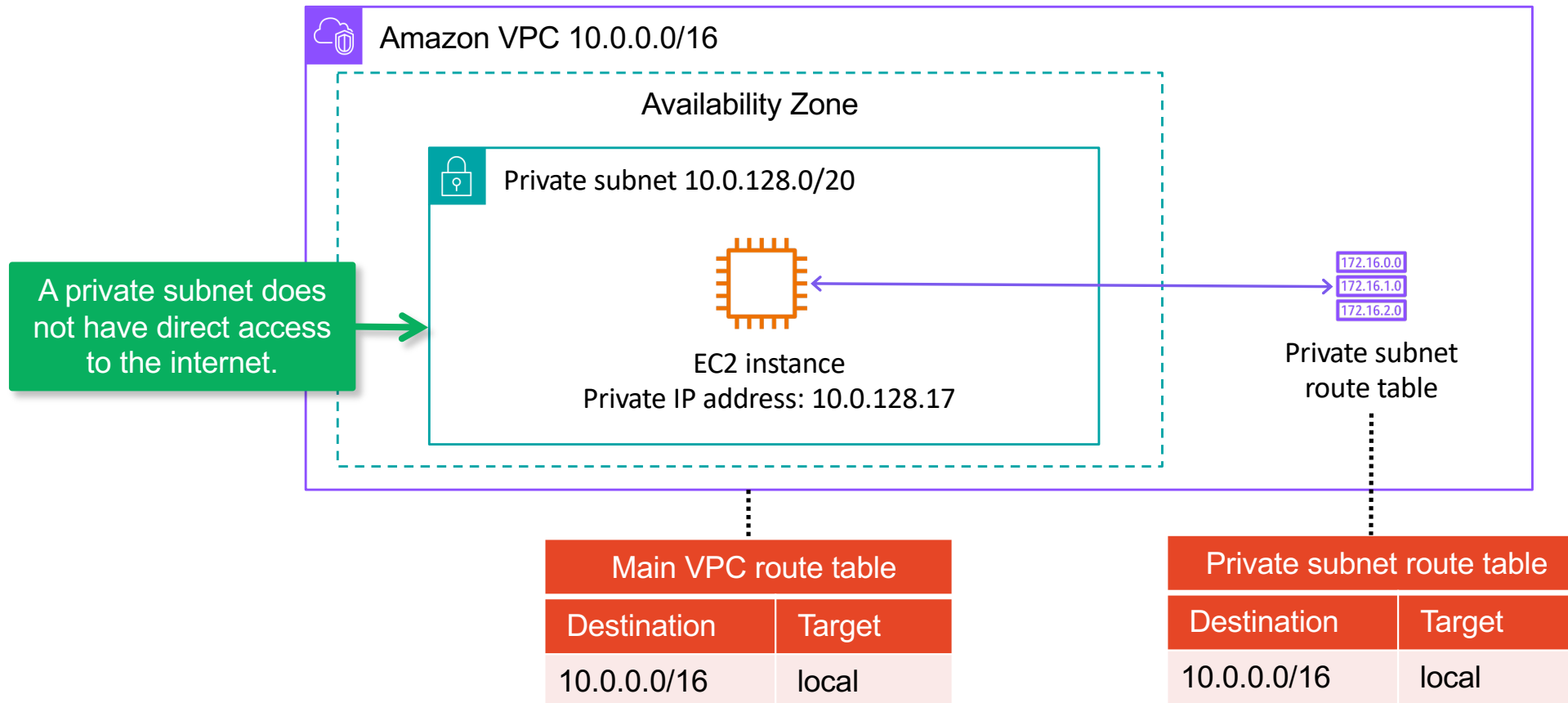
Internet Gateway

- An Internet Gateway (IGW) is a horizontally scaled, redundant, and highly available VPC component.
 - It allows communication between instances in the VPC and the internet.
- **Two Key Functions**
 - **Network Address Translation (NAT):** Translates private IP addresses in your VPC to public IP addresses, enabling instances to connect to the internet.
 - **Route Table Target:** Provides a target in the VPC route tables for internet-routable traffic.
- **One-to-One Relationship**
 - An IGW can only be attached to a single VPC at a time.
 - A VPC can only have one IGW attached to it.

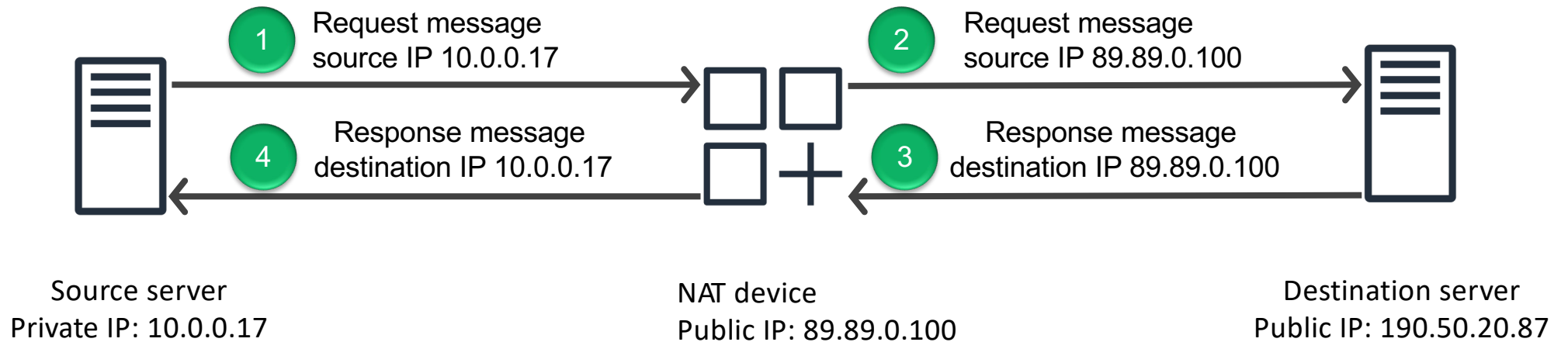
Public subnet traffic



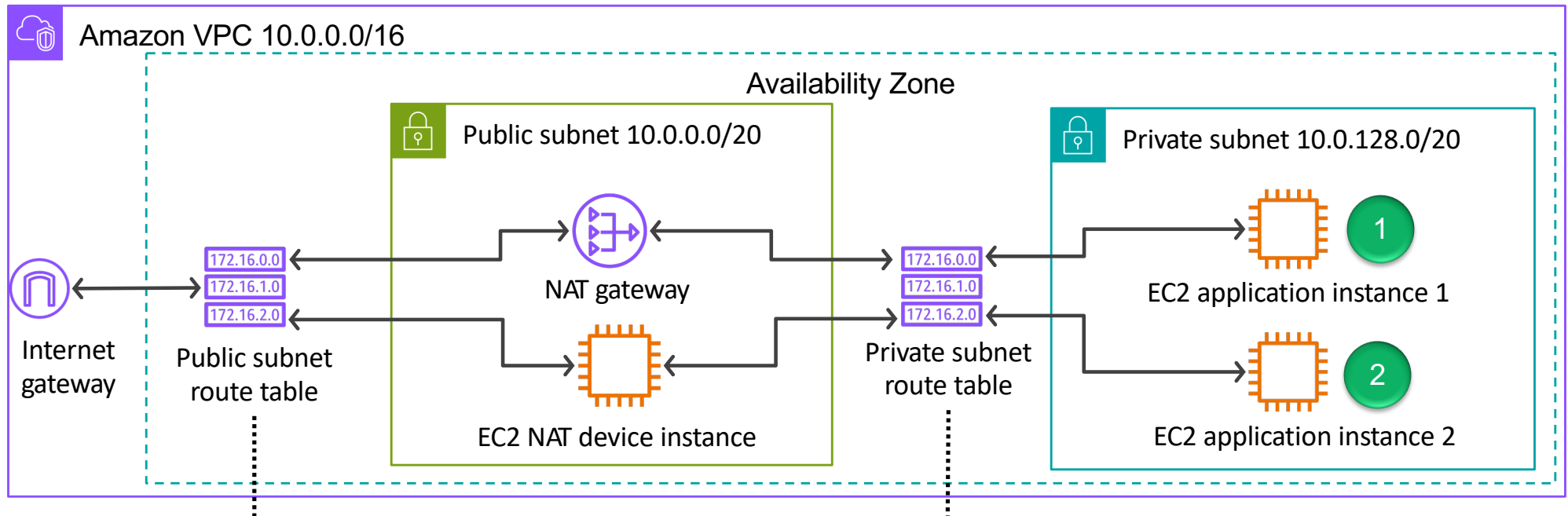
Private subnets



NAT IP mapping



Connecting private subnets to the internet



Public subnet route table	
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	Internet gateway ID

Private subnet route table	
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	NAT gateway ID

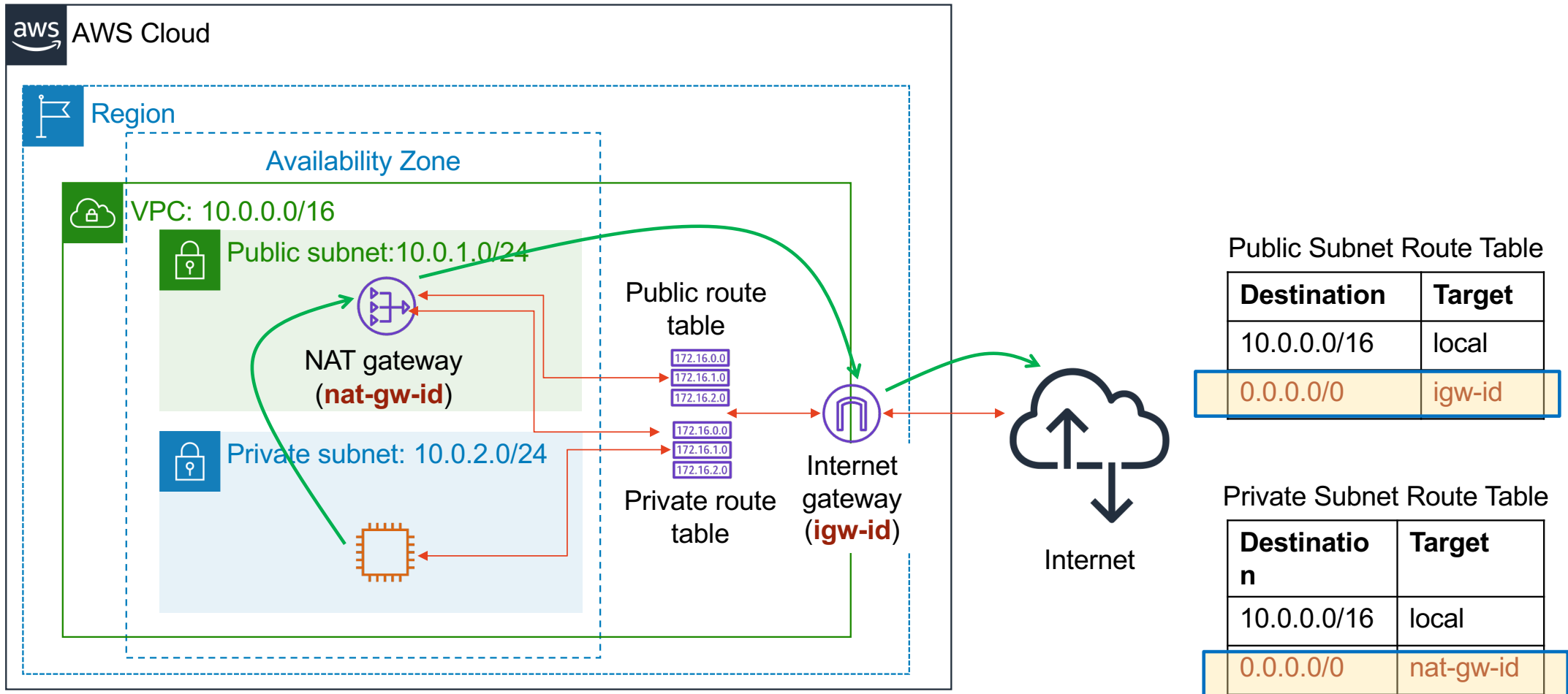
NAT gateway basics

- There are two types of NAT gateways
 - Public NAT Gateway
 - Allowing instances in private subnet to connect to Internet (outgoing only)
 - Private NAT Gateway
 - Allowing instances in private subnets to connect to other VPCs or on premises network
- Public NAT Gateway
 - Is created within a public subnet
 - Is assigned an elastic IP address
 - Has redundancy within the Availability Zone
 - Private subnets in different AZs should have their own NAT gateway

Internet gateway and NAT gateway

- A network gateway is a device or node that connects disparate networks by translating communications from one protocol to another.
- Amazon Internet Gateway (IGW) is a virtual gateway that allows instances with **public IPs** to access the internet.
 - Allows two-way traffic
 - One per VPC
- Amazon NAT Gateway (NGW) is a managed Network Address Translation (NAT) service that allows instances with **no public IPs** to access the internet.
 - Allows one-way traffic – outgoing
 - One per AZ (recommended)

Private subnet traffic



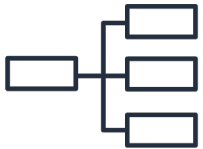
**Activity:
Choose the
Right Type
of Subnet**

- Decide whether instances should be placed into a public or private subnet.

Choose public or private subnet for each use case



Database instances



Batch-processing instances



Web application instances



NAT gateway or instance

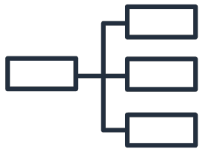
Recommended subnet selections for each use case



Database instances



Private subnet



Batch-processing instances



Private subnet



Web application instances



Public or private subnet



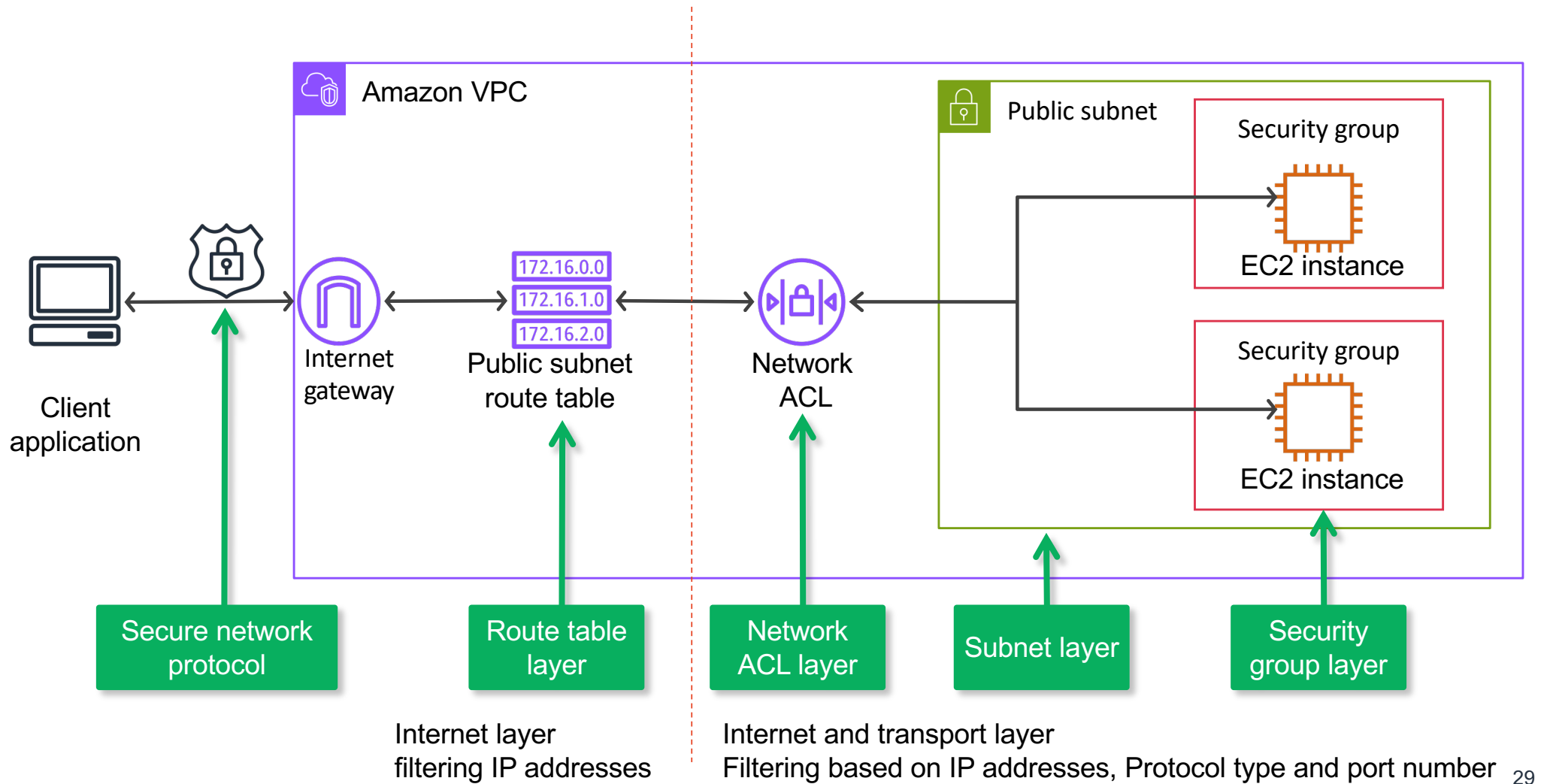
NAT gateway or instance



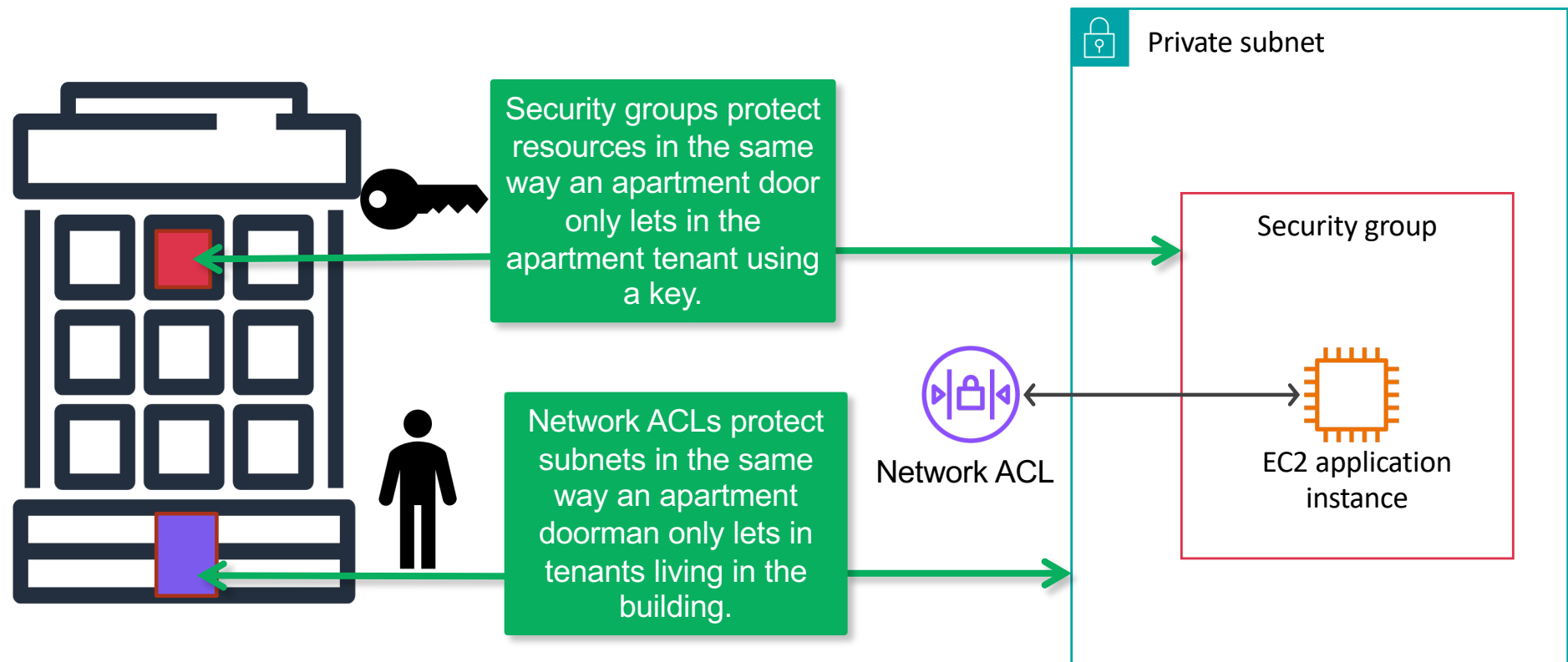
Public subnet

VPC Security

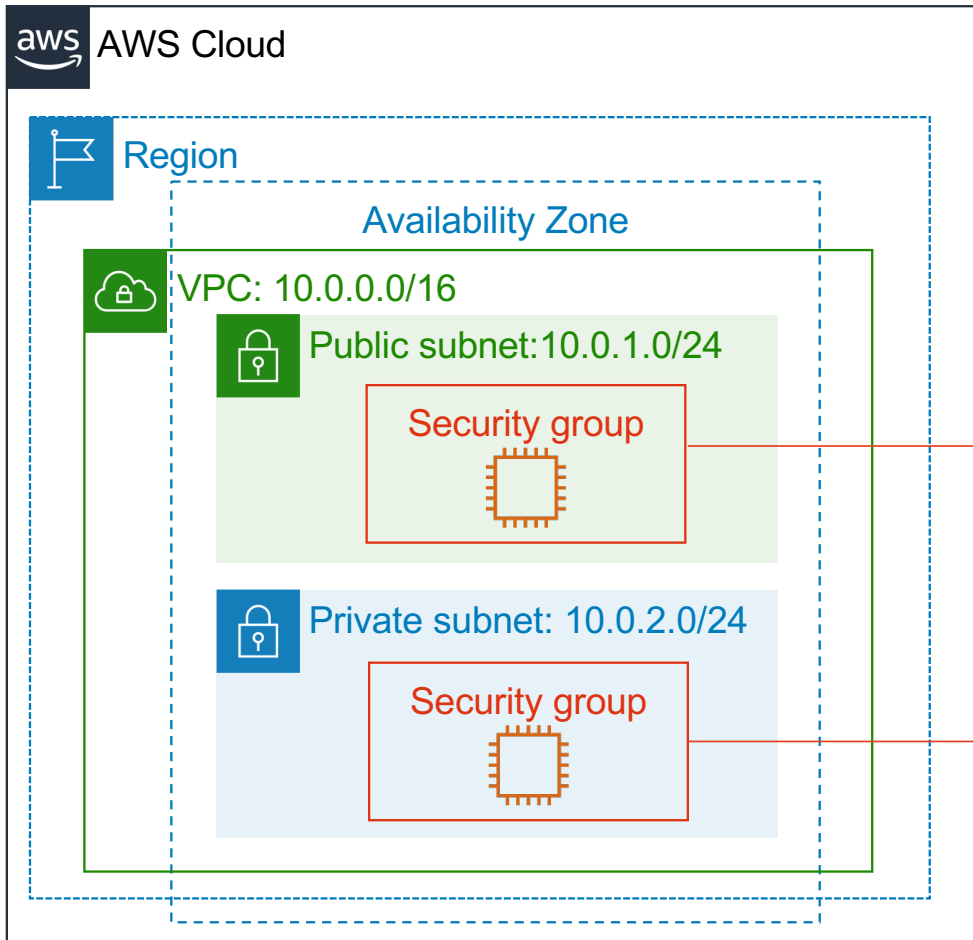
Security layers of defense



Security groups and network ACL scope

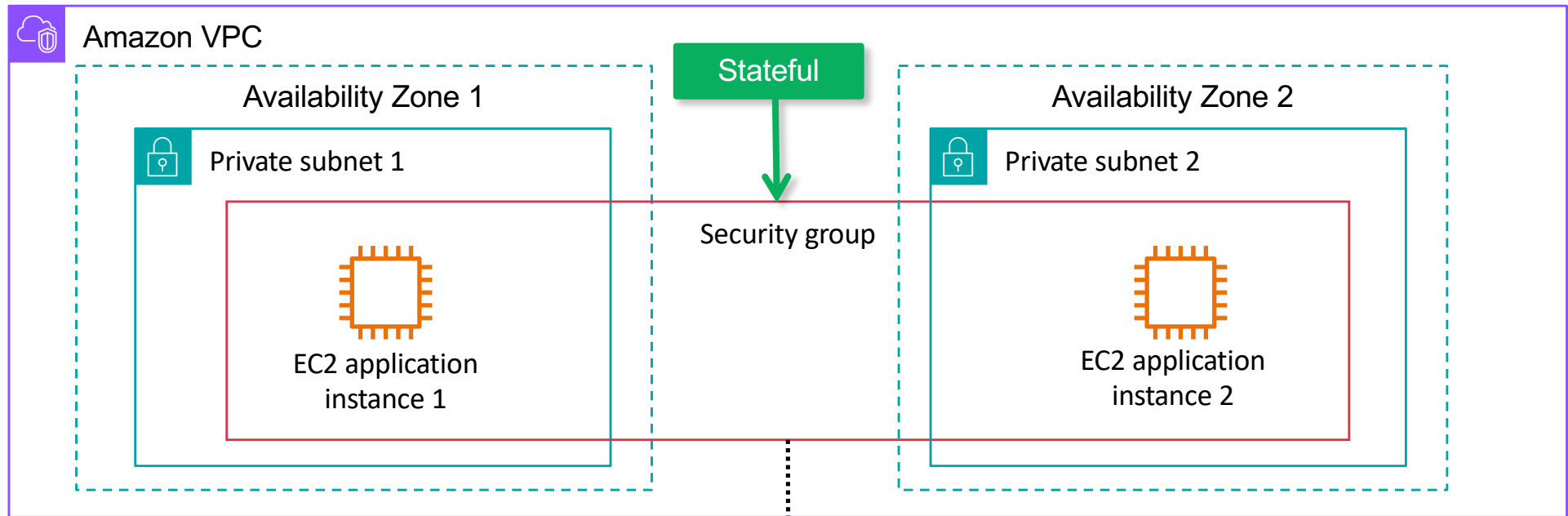


Security groups



Security groups act at the **instance level**.

Security groups

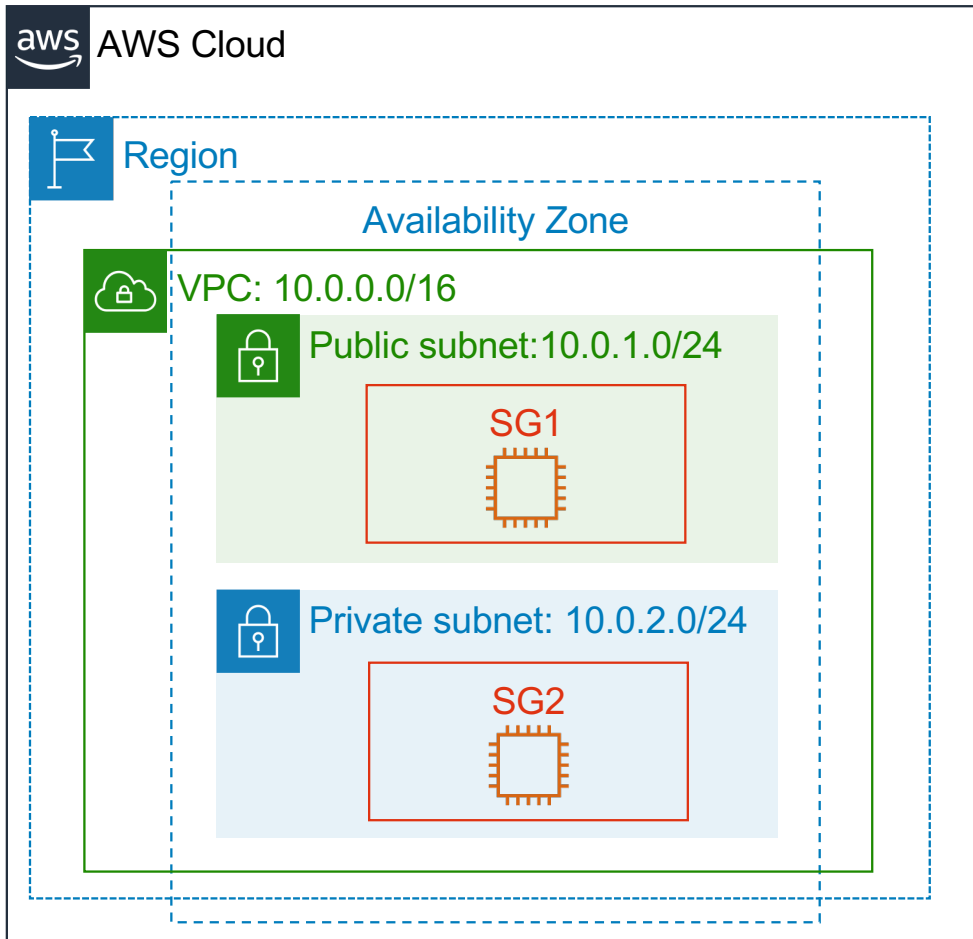


Inbound security group rule			
Source	Traffic type	Protocol	Port range
Load balancer security group ID	HTTPS	TCP	443

Security group basics

- Security groups have rules that control inbound and outbound instance traffic.
 - Each rule specifies a source/destination, protocol and port range
- All rules are *allow-rules*
 - When a new security group is created, there is no inbound rule and there is an outbound rule to allow all outbound traffic
 - No allowed inbound traffic but all outbound traffic are allowed
- Security groups are stateful.
 - Return traffic is automatically allowed, regardless of any rule

A newly created security group



SG1

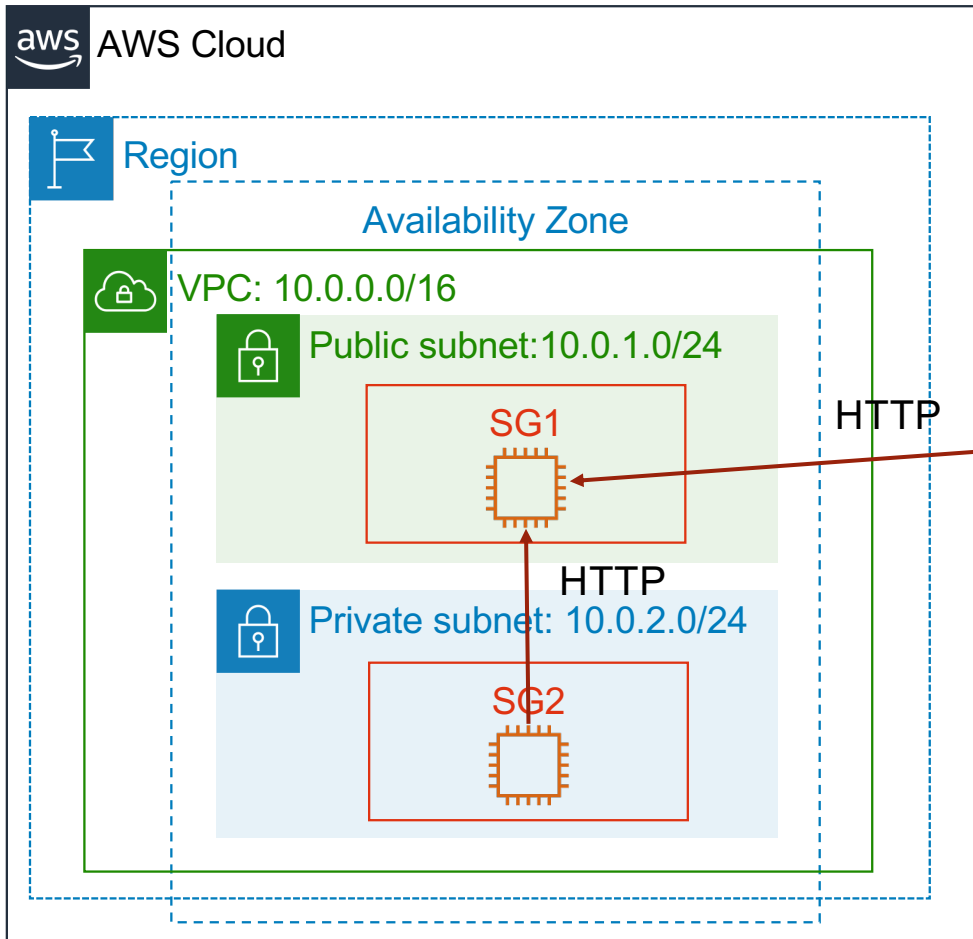
Outbound			
Destination	Protocol	Port Range	Description
0.0.0.0/0	All	All	Allow all outbound IPv4 traffic.

Instances in SG1 can browse the Internet, or download software from internet

These are considered as outbound traffic

Because SG rules are stateful, the responses of all outgoing requests are allowed to reach the instance

An incoming rule example



SG1

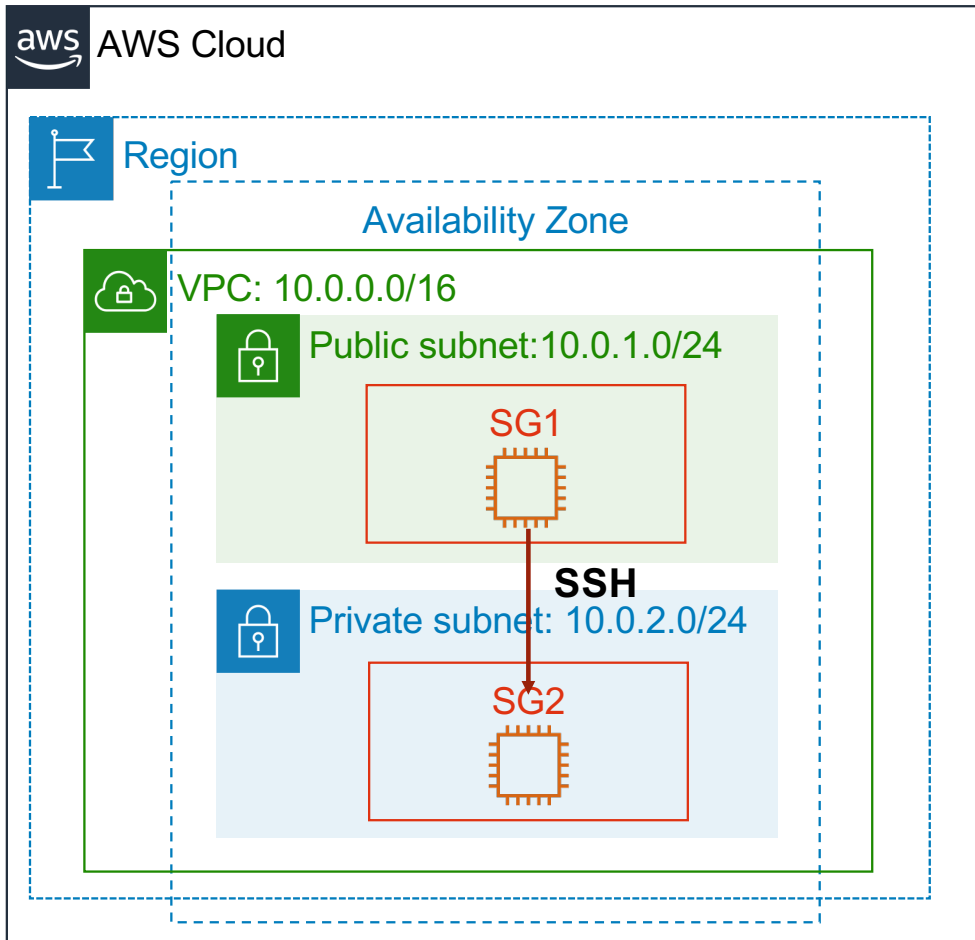
Inbound			
Source	Protocol	Port Range	Description
0.0.0.0/0	TCP	80	Allow inbound HTTP traffic.



This rule permits unrestricted HTTP traffic (port 80, TCP) from any source to any instance associated with SG1

The CIDR block 0.0.0.0/0 matches any host

Another incoming rule example

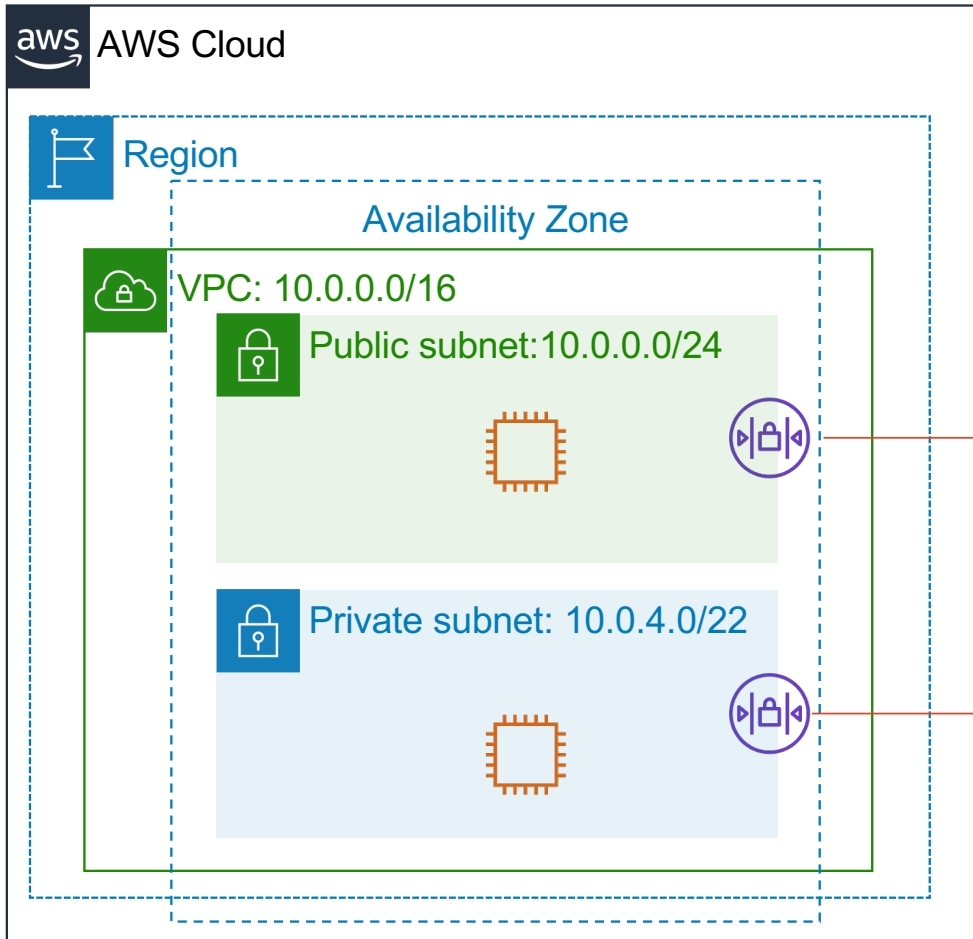


SG2

Inbound			
Source	Protocol	Port Range	Description
SG1	TCP	22	Allows SSH traffic from SG1

This rule permits SSH traffic (port 22, TCP) from any instance associated with SG1 to any instance associated with SG2

Network access control lists (network ACLs 1 of 2)



Network ACLs act at the **subnet level**.

Network access control lists (network ACLs 2 of 2)

- A network ACL has **separate inbound and outbound rules**, and each rule can either **allow or deny traffic**.
- **Default** network ACLs **allow** all inbound and outbound IPv4 traffic.
- Network ACLs are **stateless**.

Inbound						
Rule	Type	Protocol	Port Range	Source		Allow/Deny
100	All IPv4 traffic	All	All	0.0.0.0/0		ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0		DENY
Outbound						
Rule	Type	Protocol	Port Range	Destination		Allow/Deny
100	All IPv4 traffic	All	All	0.0.0.0/0		ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0		DENY

Custom network ACLs examples

- **Custom** network ACLs **deny** all inbound and outbound traffic until you add rules.
- You can specify **both allow and deny** rules.
- Rules are evaluated in number order, starting with the **lowest number**.

Inbound					
Rule	Type	Protocol	Port Range	Source	Allow/Deny
100	HTTPS	TCP	443	0.0.0.0/0	ALLOW
120	SSH	TCP	22	192.0.2.0/24	ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0	DENY
Outbound					
Rule	Type	Protocol	Port Range	Destination	Allow/Deny
100	HTTPS	TCP	443	0.0.0.0/0	ALLOW
120	SSH	TCP	22	192.0.2.0/24	ALLOW
*	All IPv4 traffic	All	All	0.0.0.0/0	DENY

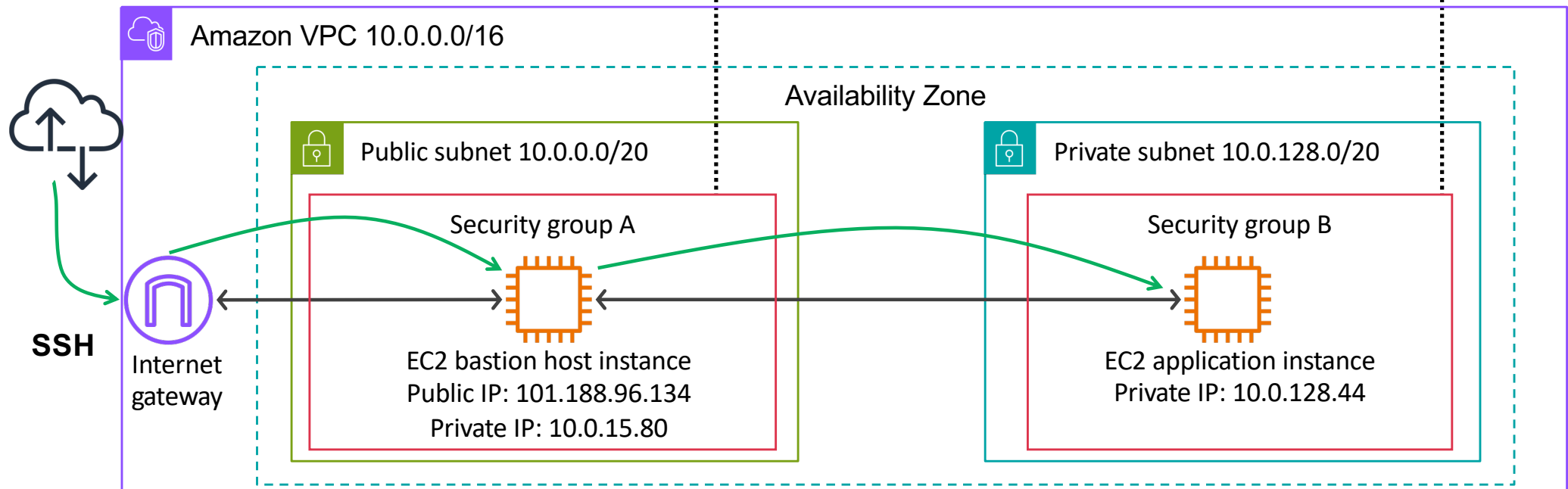
Security groups versus network ACLs

Attribute	Security Groups	Network ACLs
Scope	Instance level	Subnet level
Supported Rules	Allow rules only	Allow and deny rules
State	Stateful (return traffic is automatically allowed, regardless of rules)	Stateless (return traffic must be explicitly allowed by rules)
Order of Rules	All rules are evaluated before decision to allow traffic	Rules are evaluated in number order before decision to allow traffic
Default rules	No inbound traffic is allowed All outbound traffic is allowed	All inbound traffic is allowed All outbound traffic is allowed

Administering resources with bastion hosts

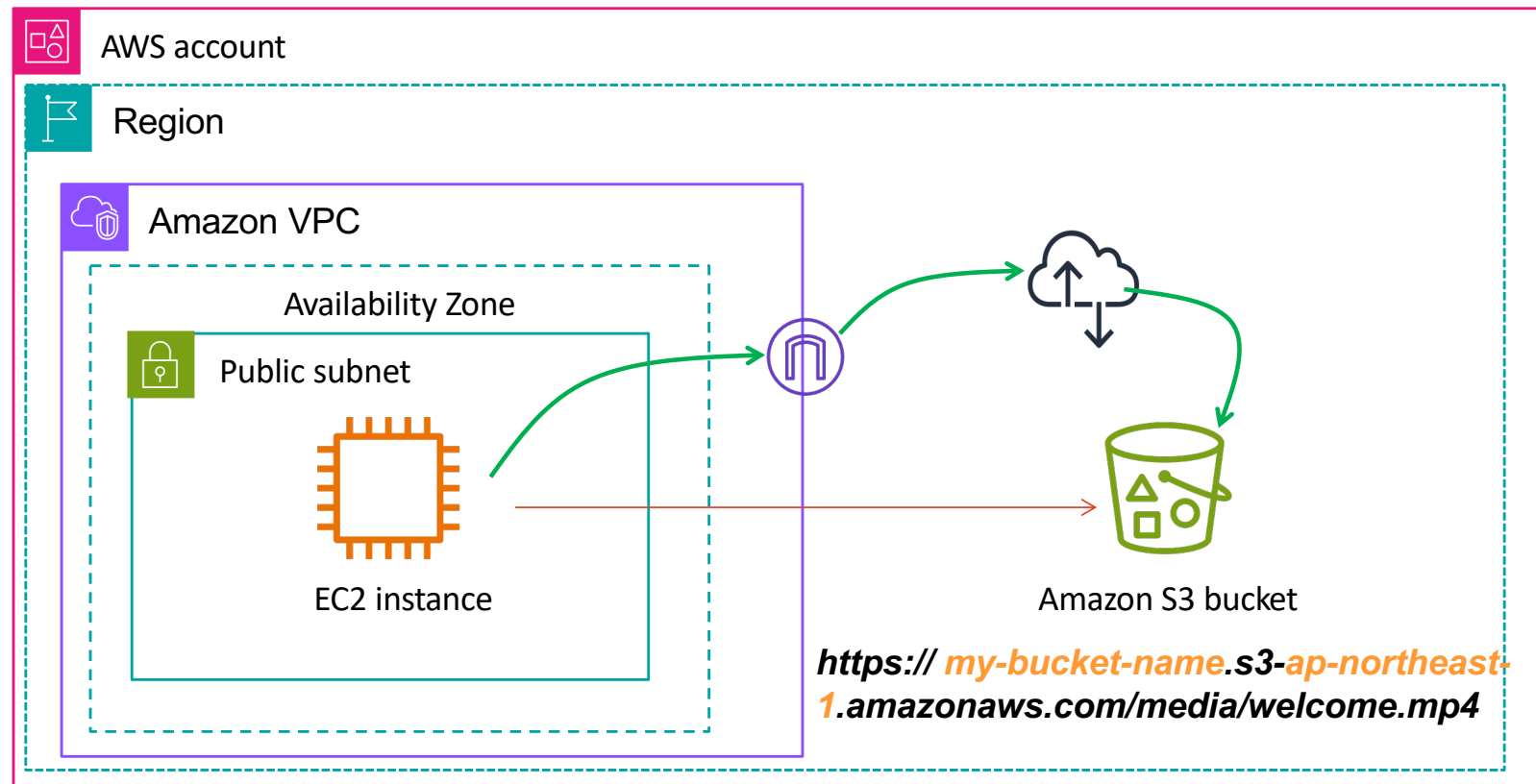
Security Group A inbound rule			
Source	Type	Protocol	Port range
IP address range	SSH	TCP	22

Security Group B inbound rule			
Source	Traffic type	Protocol	Port range
Security group A	SSH	TCP	22

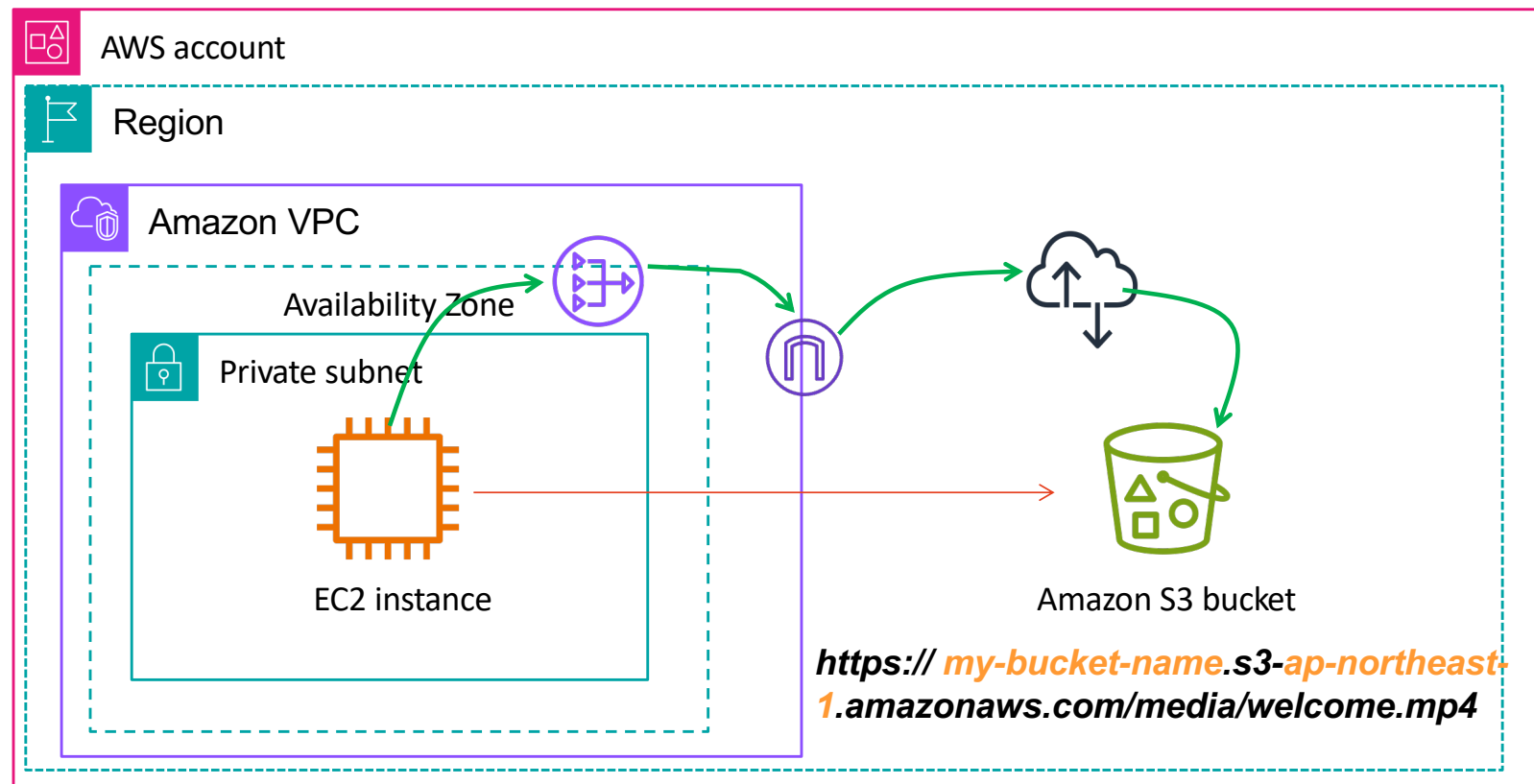


Connecting to managed AWS services

How to connect to managed AWS services (public subnet)



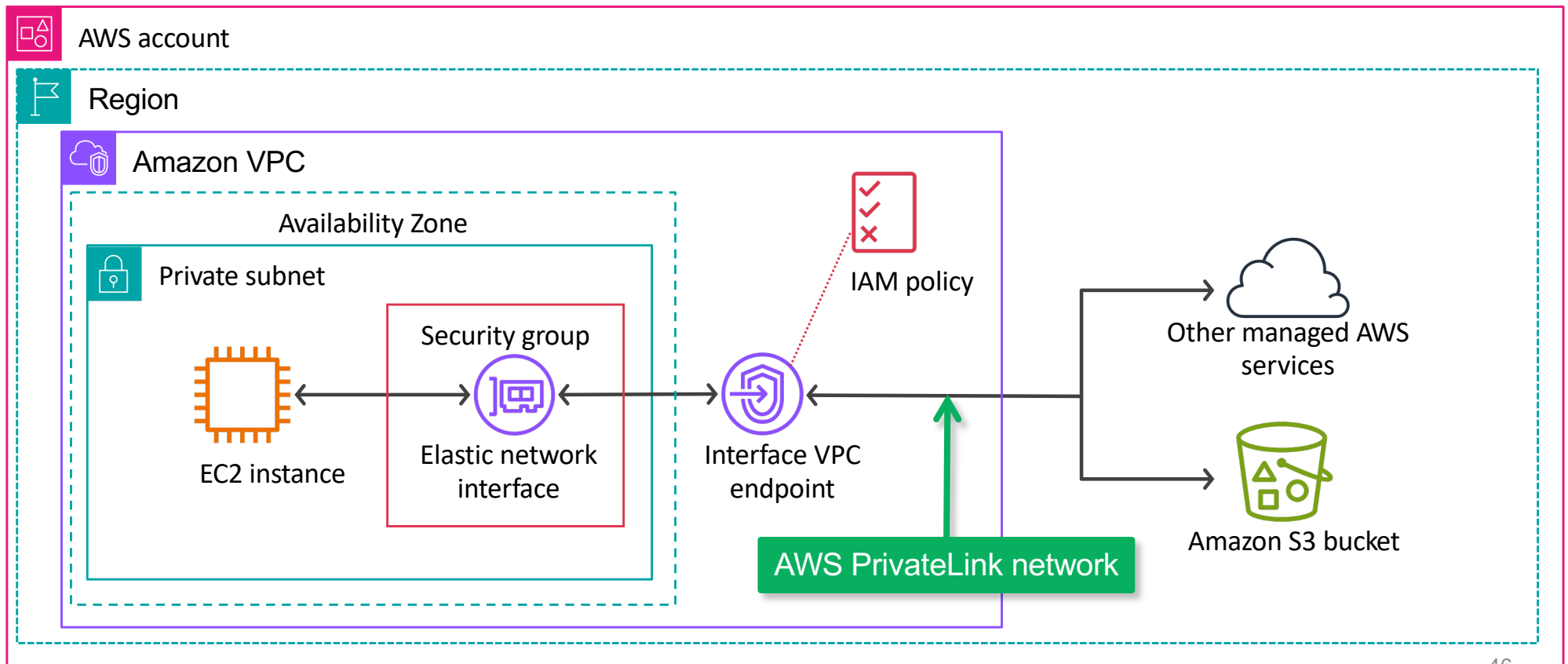
How to connect to managed AWS services (private subnet)



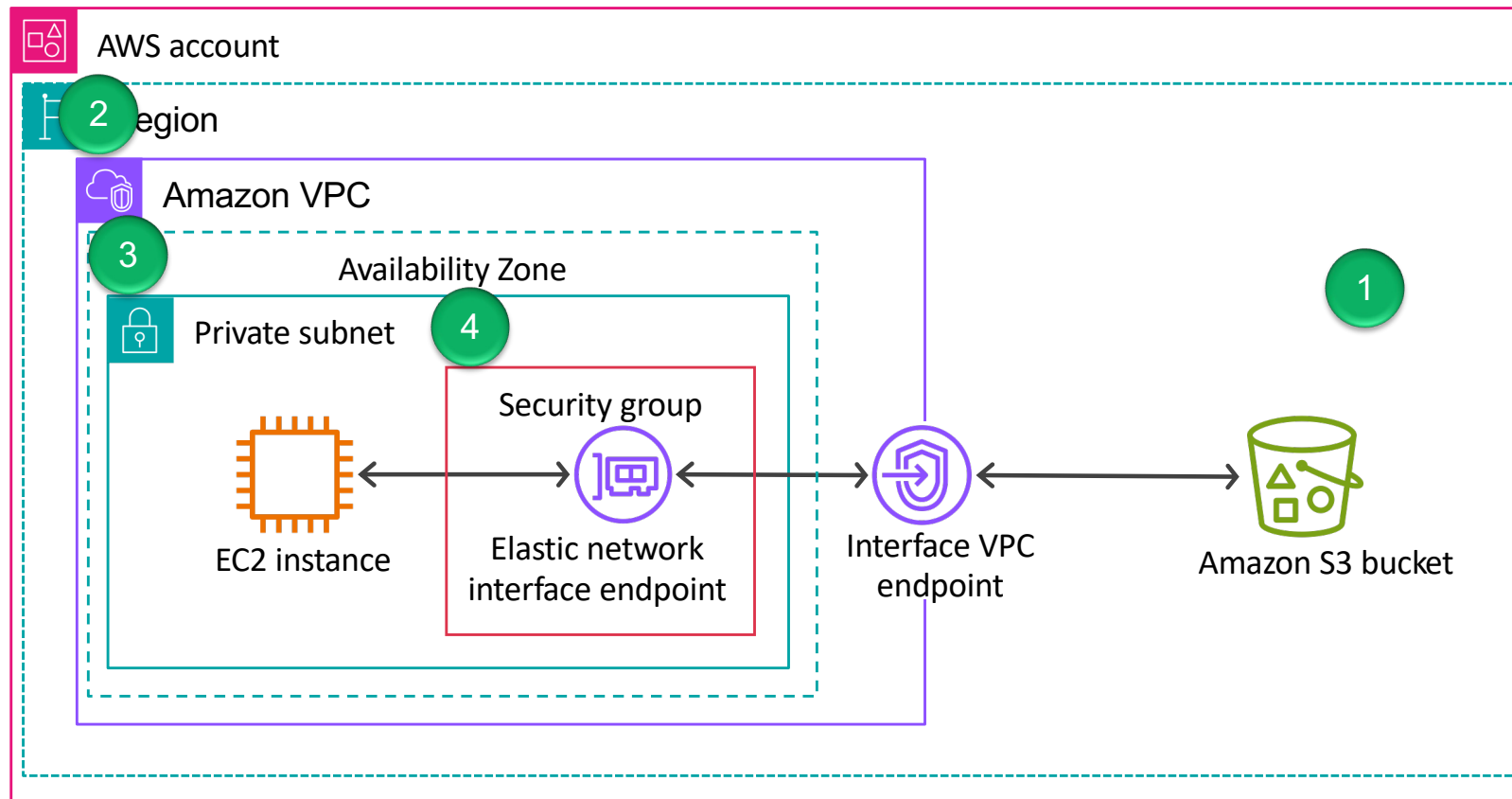
VPC Endpoints

- **VPC Endpoint** allows private connection between an VPC and AWS services without going through the internet.
- It helps improve **security** and **performance**, as well as reducing potential networking cost
- Two types:
 - **Interface Endpoint**
 - Powered by **PrivateLink**
 - Uses **Elastic Network Interfaces (ENIs)**
 - **Gateway Endpoint**
 - Targeted at **S3** and **DynamoDB**
 - Adds a route in your **route table**

Interface VPC endpoints



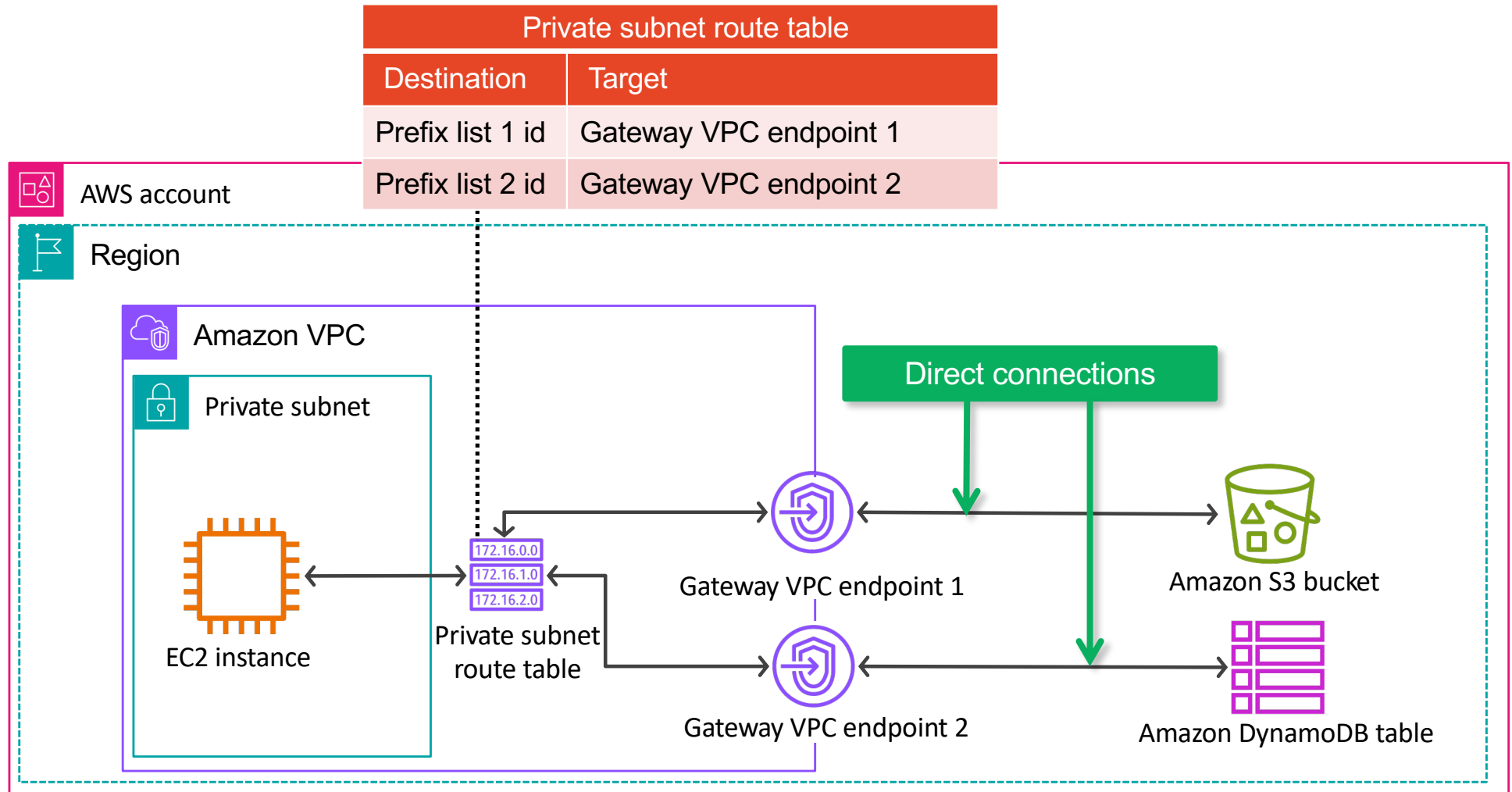
How to set up an interface VPC endpoint



EC2 attached ENI vs. VPC Endpoint ENI

- Both are backed by Nitro hardware
- EC2-attached ENIs
 - **Backed by Nitro hardware** — specifically the **Nitro card for VPC**.
 - Handles **VPC networking, security groups, monitoring, and encryption** offloaded from the host CPU.
- ENIs for interface VPC endpoints
 - Also backed by Nitro hardware
 - **These ENIs are not attached to an EC2 instance, but AWS still provisions them on Nitro-backed hypervisors running in AWS-managed infrastructure.**
 - These are “invisible” **service instances**, purpose-built to act as **entry points** to AWS services via **PrivateLink**.

Gateway VPC endpoints



Gateways and VPC Endpoints

“A gateway connects your VPC to another network. For example, use an internet gateway to connect your VPC to the internet. Use a VPC endpoint to connect to AWS services privately, without the use of an internet gateway or NAT device”

Gateways vs. VPC Endpoints

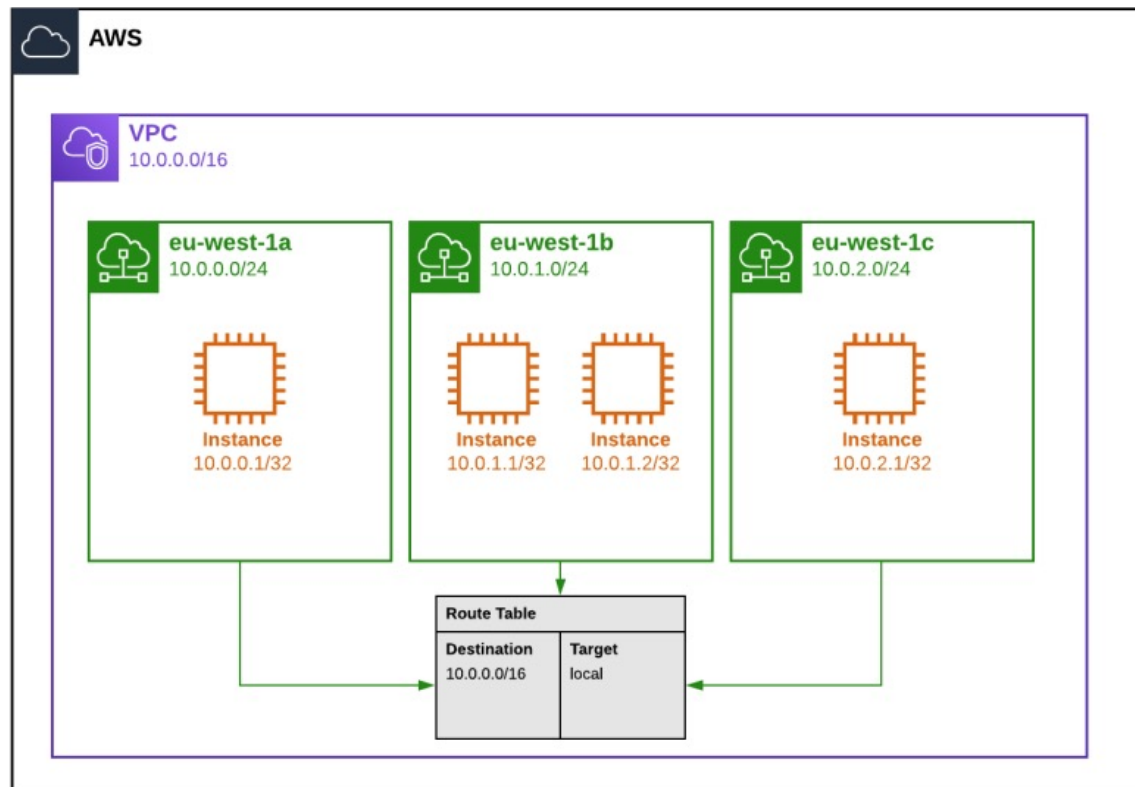
Feature	Gateway	VPC Endpoint
Purpose	Connect to external networks (internet, other VPCs)	Connect privately to AWS services
Services	Internet Gateway (2-way) / NAT Gateway(outbound only)	Interface/Gateway Endpoint
Traffic Type	Leaves AWS network	Stays in AWS network
Traffic direction	Inbound and outbound	Primarily outbound to AWS services
Cost	NAT Gateway: \$\$, Internet Gateway: free	Endpoints: low cost (data processing fee)

AWS Internal Traffic

routing for VPC destination

Based on <https://www.sentiablog.com/amazon-vpc-the-picasso-of-software-defined-networking>

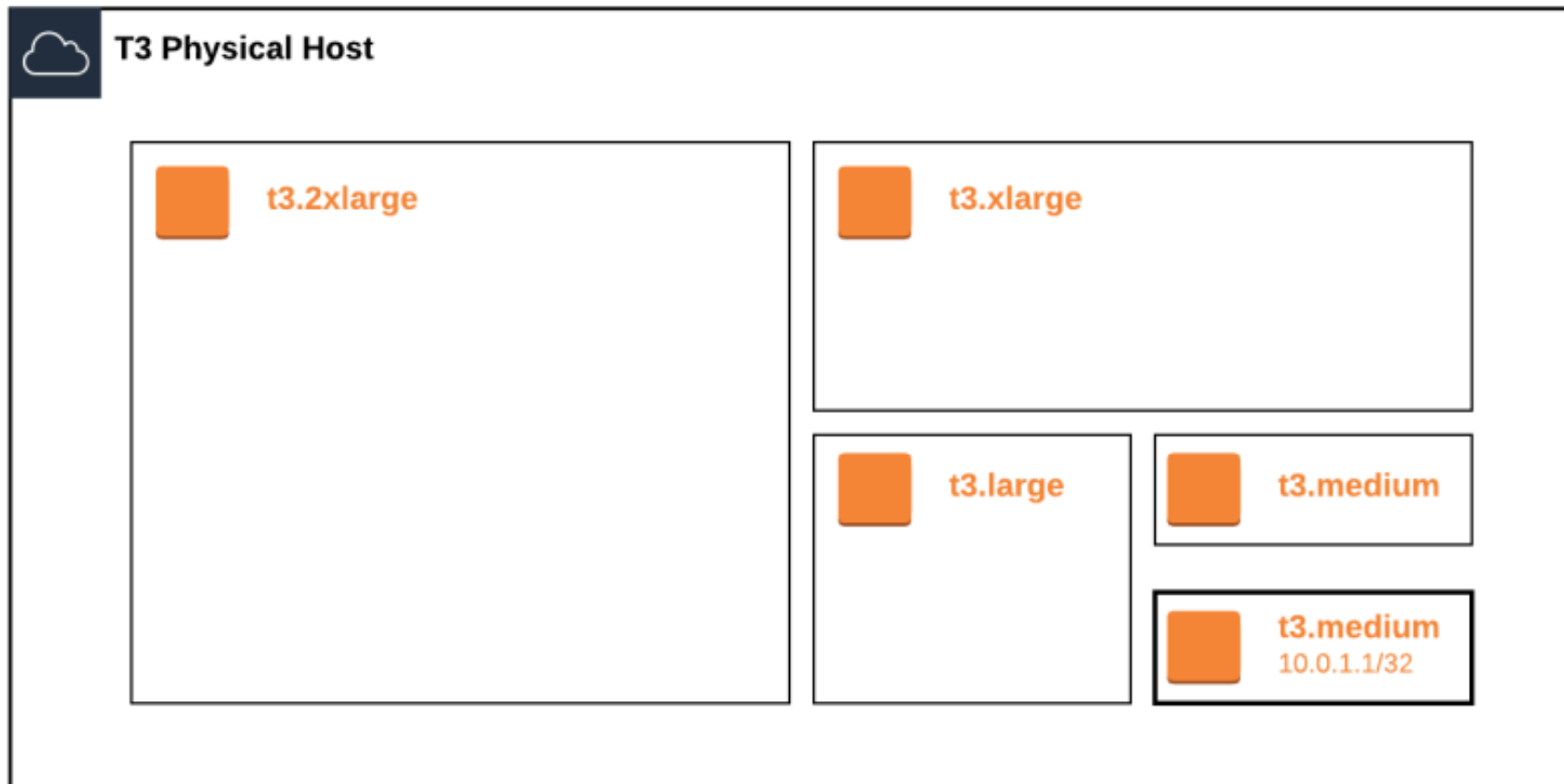
VPC and EC2 instances



This is the default main route table of a custom VPC

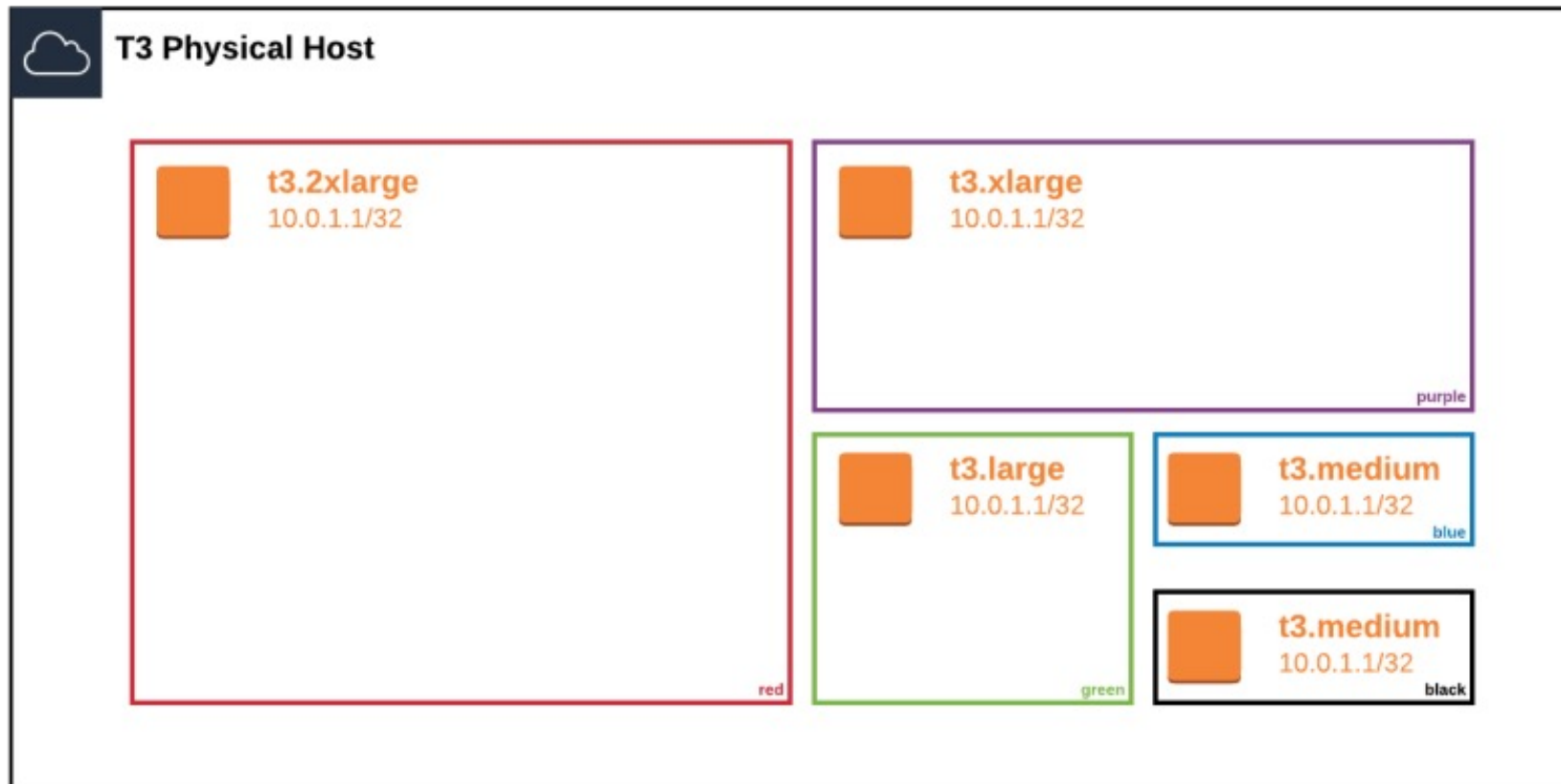
It allows instances within the same VPC to communicate with each other

EC2 and Physical Host

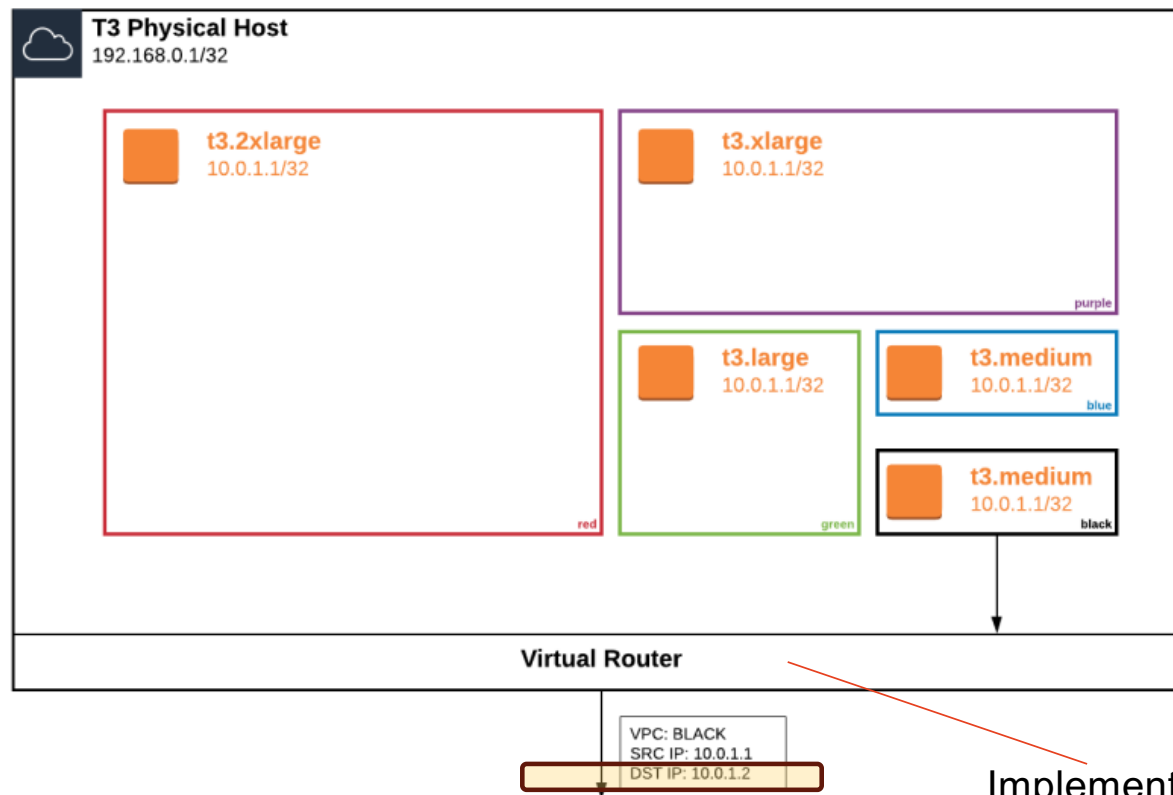


Multitenant physical host

Instances in different VPCs can have the same private IP address

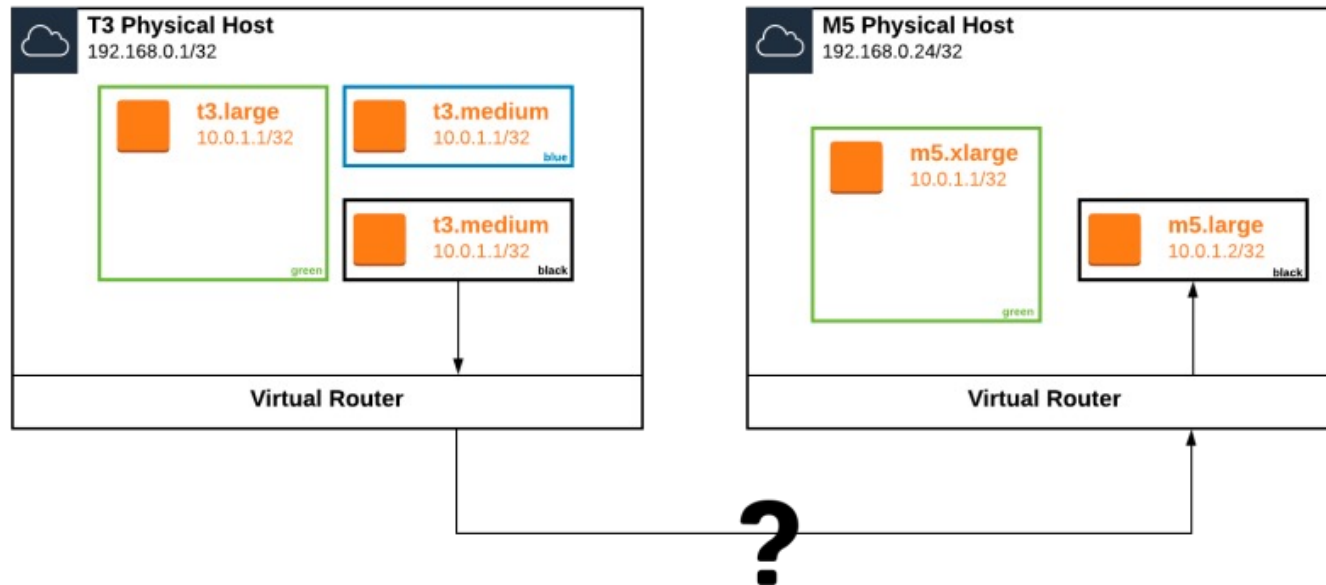


Virtual router and encapsulation

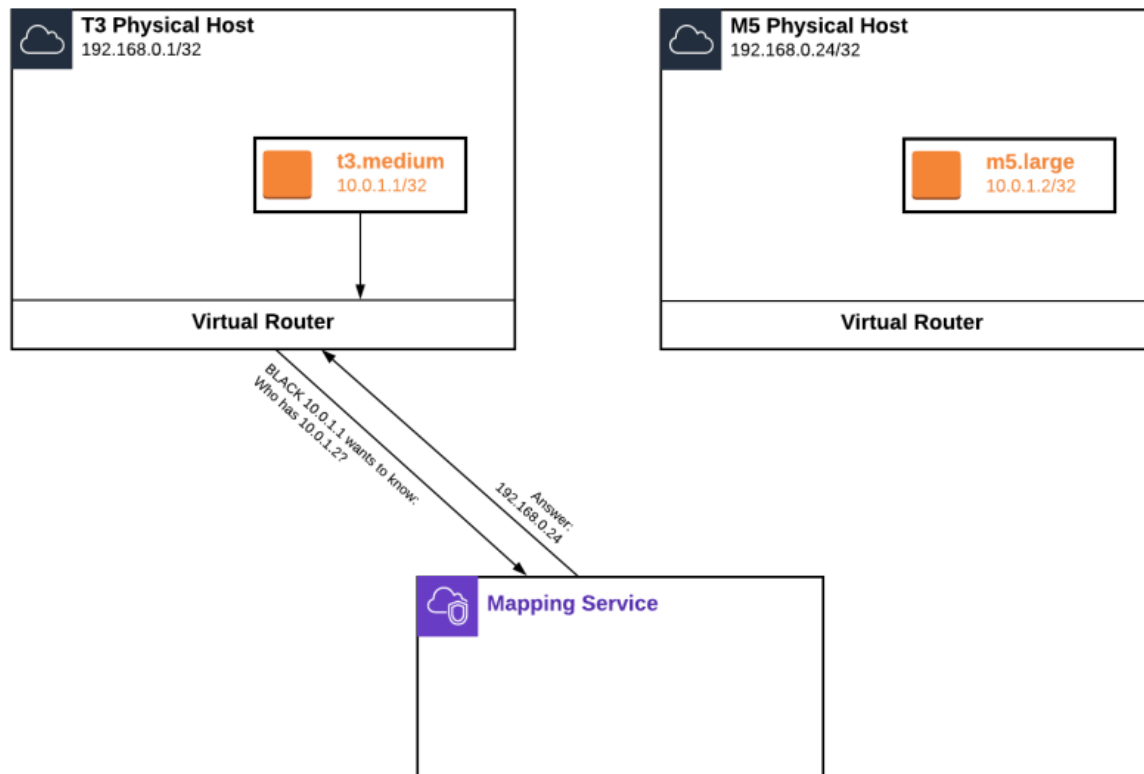


Implemented by Nitro card for VPC

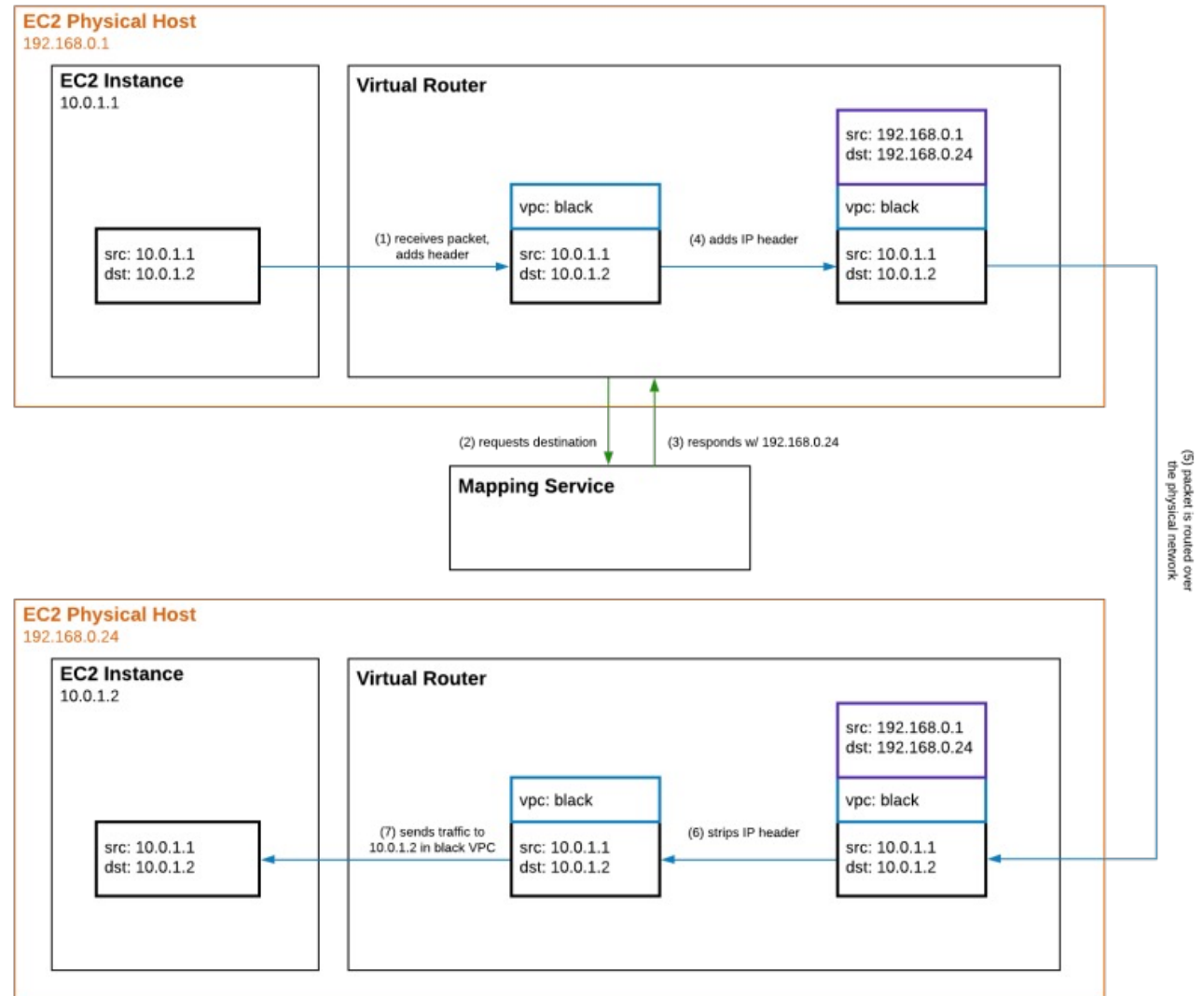
The mapping service



The mapping service



The actual traffic

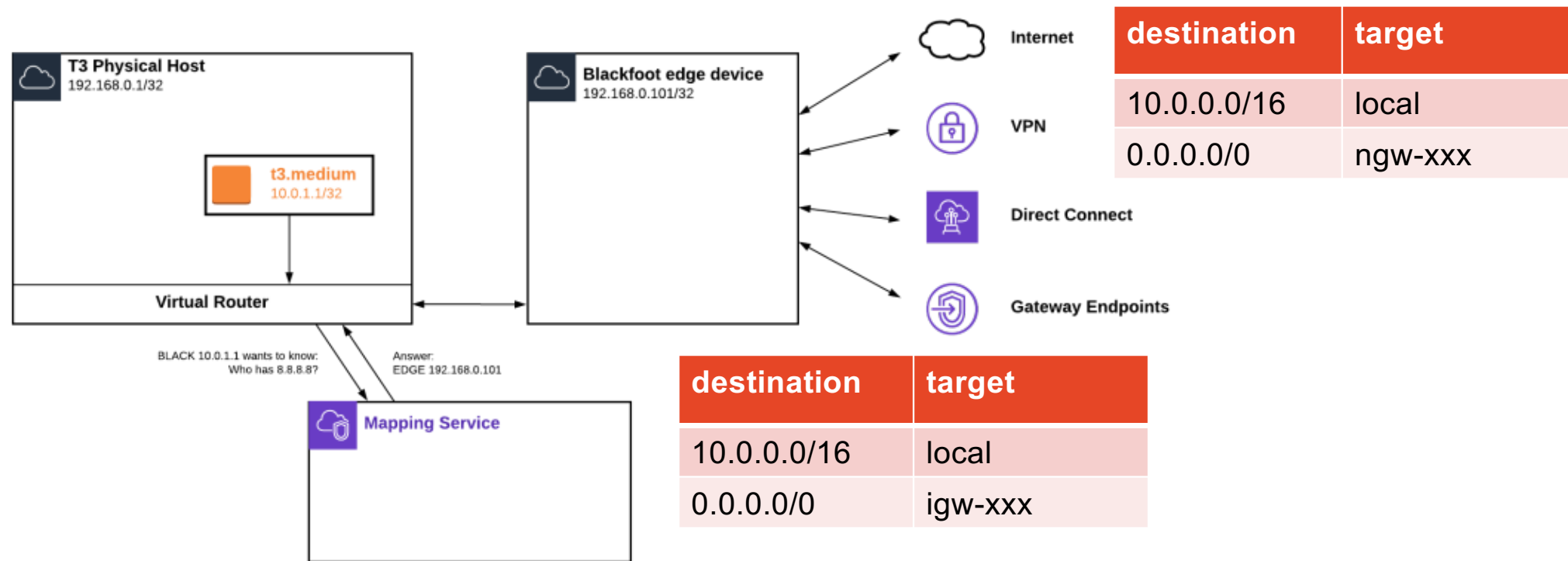


AWS External Traffic

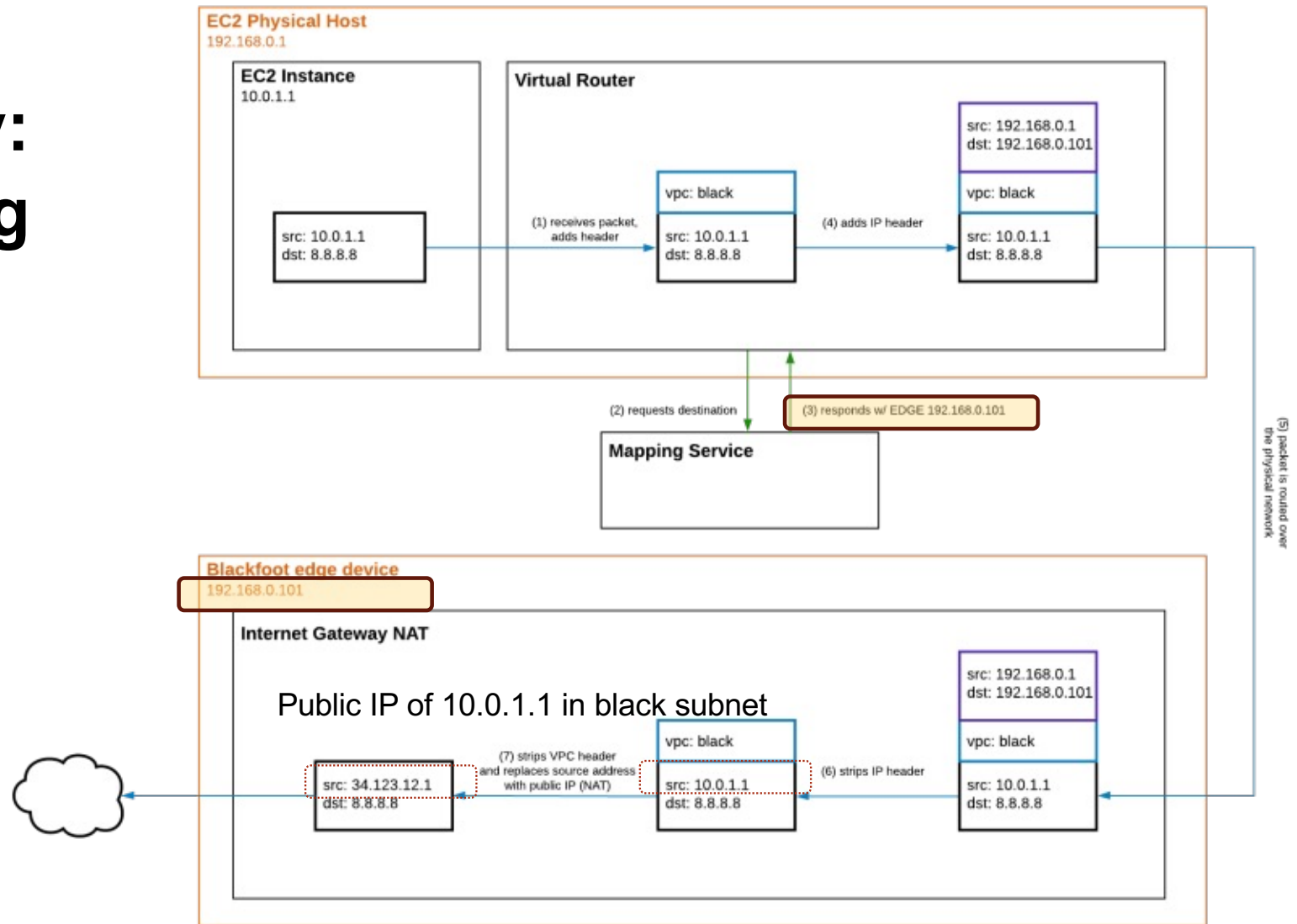
routing for non-VPC destination

Based on <https://www.sentiablog.com/amazon-vpc-the-picasso-of-software-defined-networking>

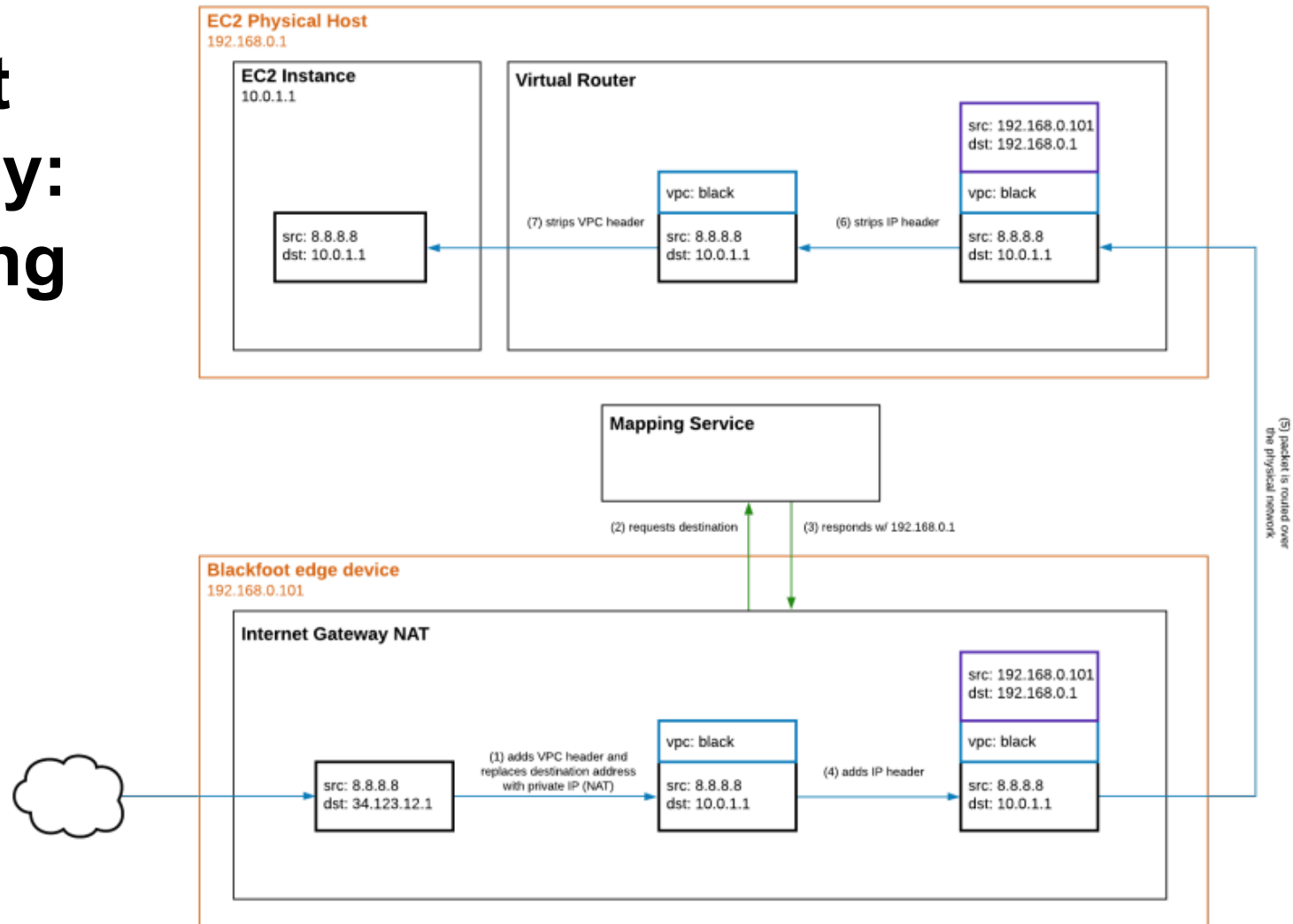
Non-VPC destinations

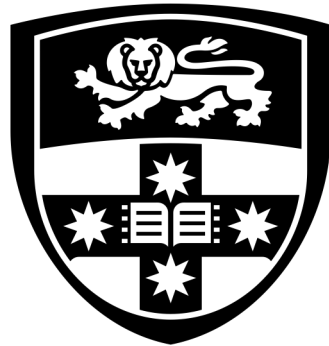


Internet Gateway: Outgoing



Internet Gateway: Incoming





THE UNIVERSITY OF
SYDNEY