

AWS Academy Cloud Architecting

Module 6: Creating a Networking Environment



Sections

1. Architectural need
2. Creating an AWS networking environment
3. Connecting your AWS networking environment to the internet
4. Securing your AWS networking environment

Demonstration

- Creating a Virtual Private Cloud

Labs

- Guided Lab: Creating a Virtual Private Cloud
- **Challenge Lab:**



Knowledge check

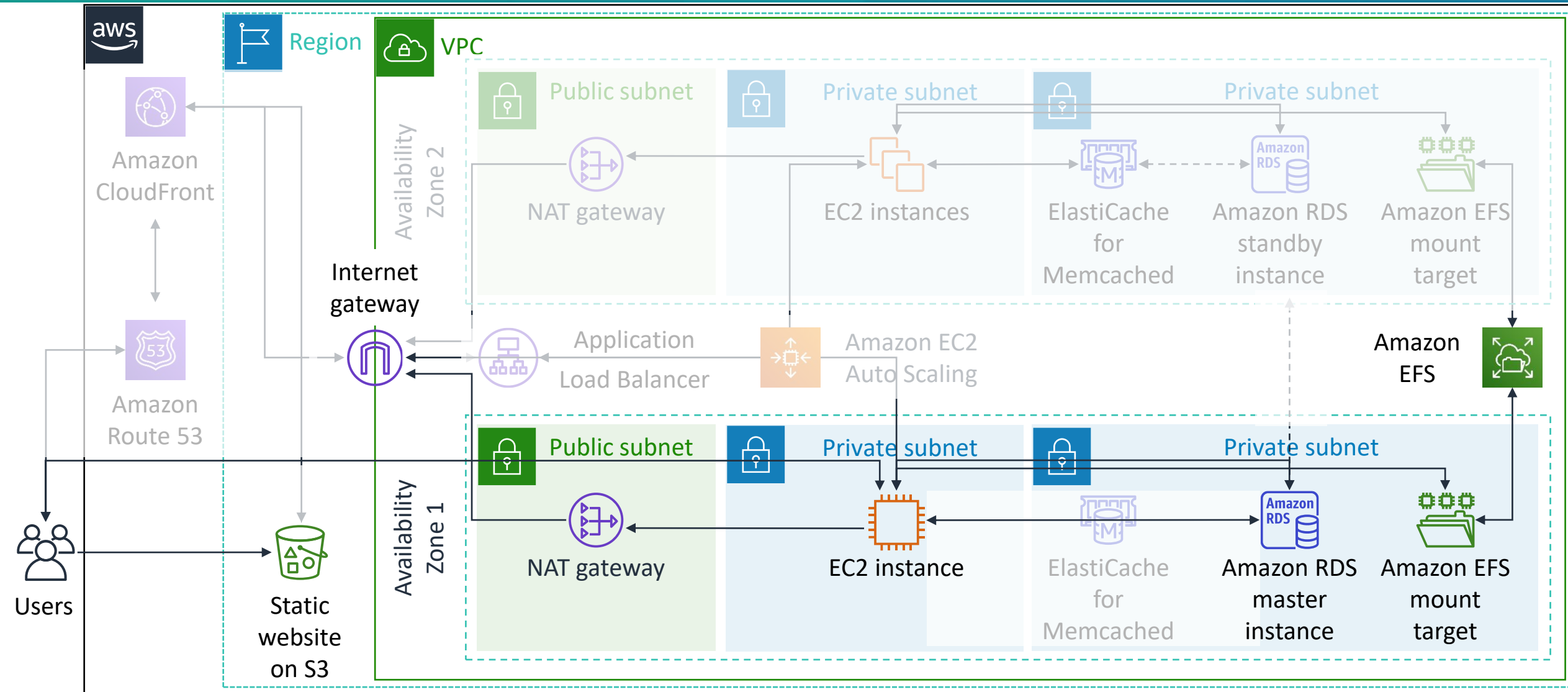
At the end of this module, you should be able to:

- Explain the foundational role of a virtual private cloud (VPC) in Amazon Web Services (AWS) Cloud networking
- Identify how to connect your AWS networking environment to the internet
- Describe how to isolate resources within your AWS networking environment
- Create a VPC with subnets, an internet gateway, route tables, and a security group

Module 6: Creating a Networking Environment

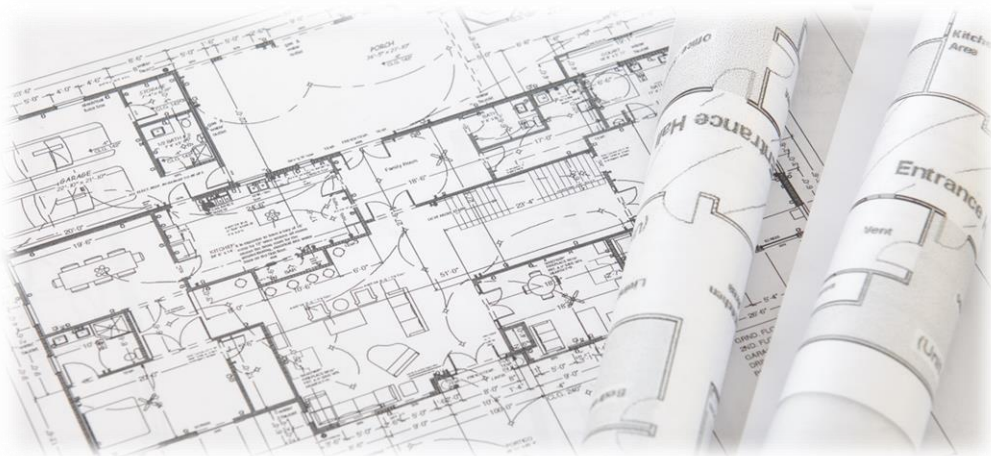
Section 1: Architectural need

Networking as part of a larger architecture



Café business requirement

The café must deploy and manage AWS resources in a secure, isolated network environment.

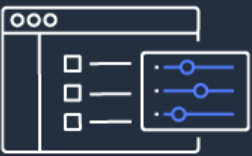


Module 6: Creating a Networking Environment

Section 2: Creating an AWS networking environment

Provision a **logically isolated section** of the AWS Cloud where you can launch AWS resources in a **virtual network that you define**.

Bring your own network



IP Addresses



Subnets



Routing rules

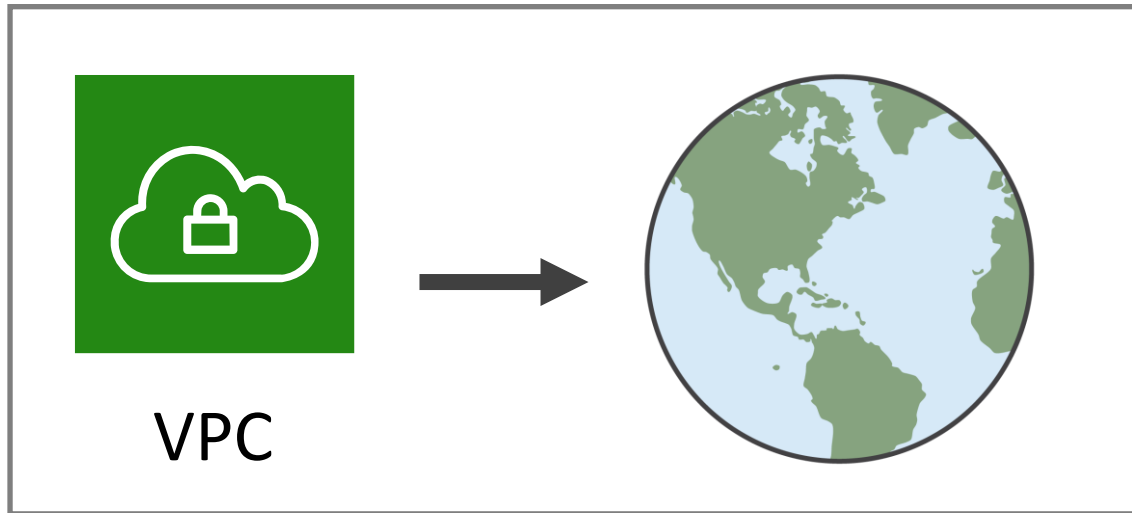


Network
configuration

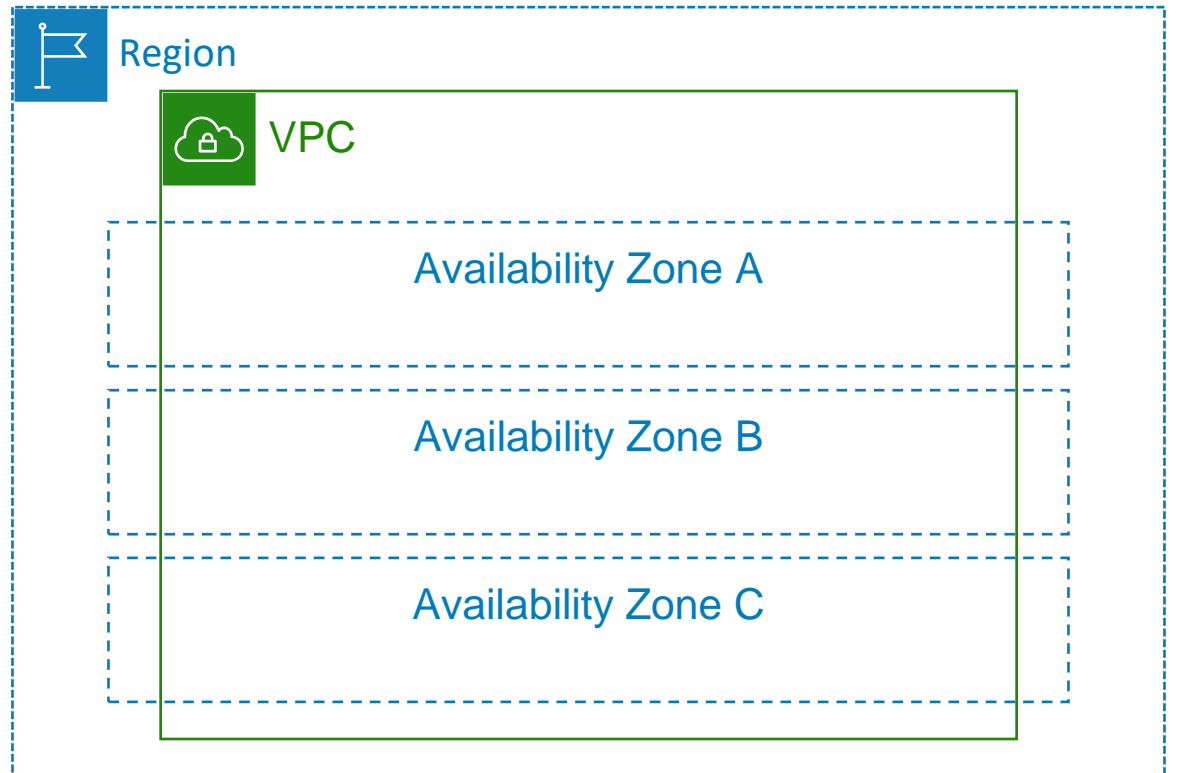


Security rules

VPC deployment



You can deploy a VPC in any AWS Region.



A VPC can host supported resources from any Availability Zone within its Region.

Classless Inter-Domain Routing (CIDR)

0.0.0.0/0 = All IP addresses

10.22.33.44/32 = 10.22.33.44

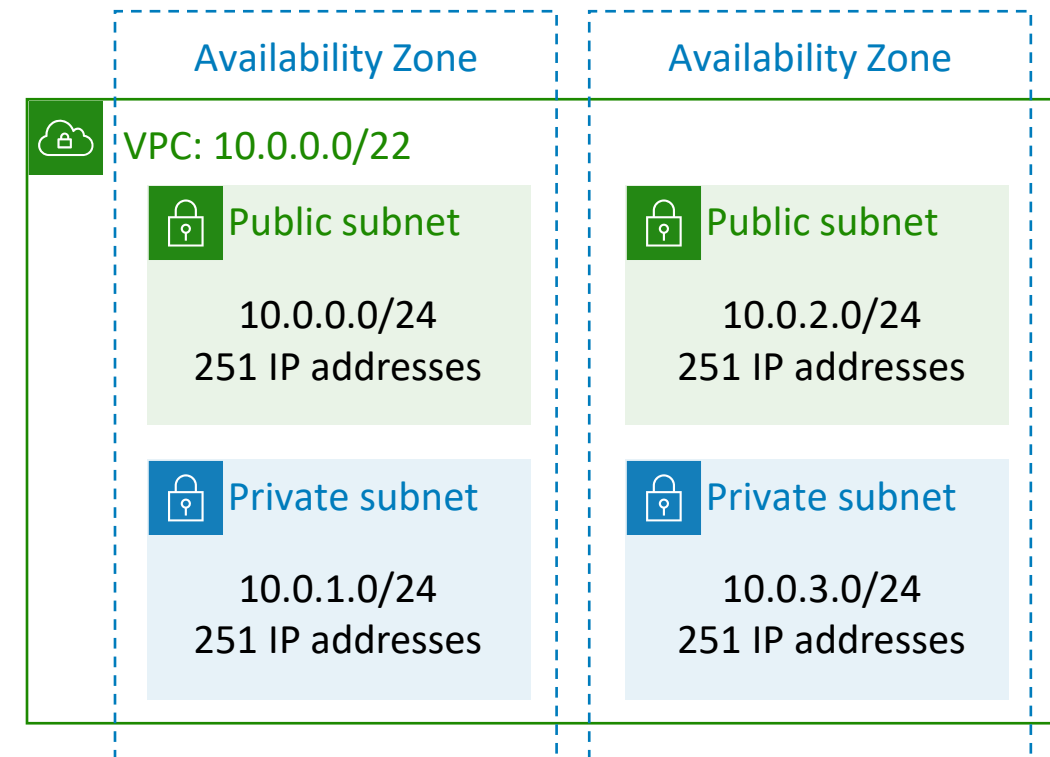
10.22.33.0/24 = 10.22.33.*

10.22.0.0/16 = 10.22.*.*

CIDR	Total IP addresses
/28	16
...	...
/20	4,096
/19	8,192
/18	16,384
/17	32,768
/16	65,536

Subnets: Dividing your VPC

- A **subnet** is a segment or partition of a VPC's IP address range where you can allocate a group of resources
- Subnets are **not isolation boundaries**
- Subnets are a **subset** of the VPC CIDR block
- Subnet CIDR blocks **cannot overlap**
- Each subnet resides entirely within one Availability Zone
- You can add one or more subnets in each Availability Zone or in a Local Zone
- AWS **reserves five IP addresses** in each subnet



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

- Create **one subnet** per available Availability Zone for **each group of hosts** that have unique routing requirements.
- **Divide your VPC network range evenly** across all available Availability Zones in a Region.
- Do not allocate all network addresses at once. Instead, ensure that you **reserve some address space** for future use.
- Size your VPC CIDR and subnets to **support significant growth** for the expected workloads.
- Ensure that your VPC network range (CIDR block) **does not overlap** with your organization's other private network ranges.

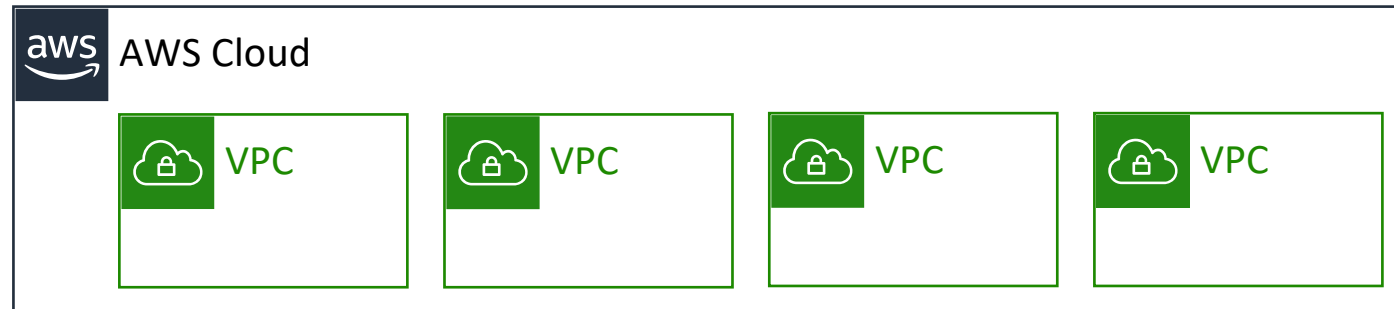
There are limited use cases where deploying **one VPC** might be appropriate:

- Small, single applications managed by a small team
- High performance computing (HPC)
- Identity management

For **most** use cases, there are two primary patterns for organizing your infrastructure: multi-VPC and multi-account.

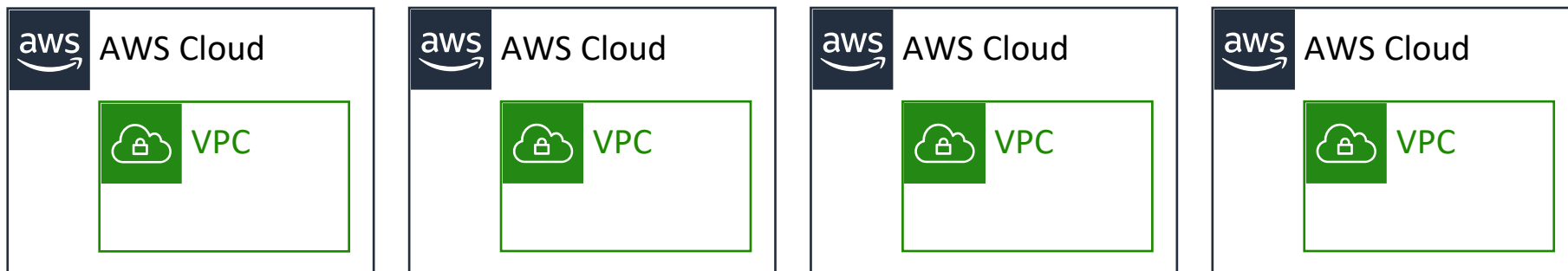
Multiple VPCs

- Best suited for –
 - **Single team** or **single organizations**, such as managed service providers
 - Limited teams, which makes it easier to **maintain standards** and **manage access**
- Exception –
 - **Governance** and **compliance standards** might require greater workload isolation regardless of organizational complexity



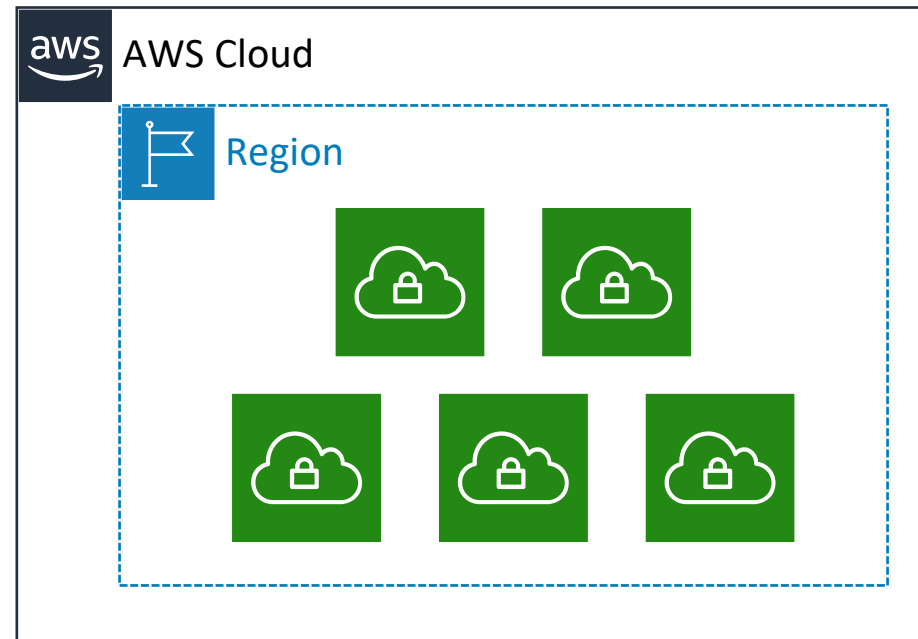
Multiple accounts

- Best suited for –
 - Large organizations and organizations with multiple IT teams
 - Medium-sized organizations that anticipate rapid growth
- Why?
 - It can be more challenging to manage access and standards in more complex organizations



Amazon VPC service limits

Service limit: 5 VPCs per Region per account



Section 2 key takeaways

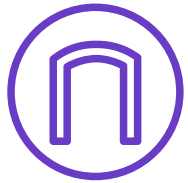


- Amazon VPC enables you to provision VPCs, which are **logically isolated sections of the AWS Cloud** where you can launch your AWS resources.
- A VPC belongs to only one Region and is divided into subnets.
- A subnet belongs to one Availability Zone or Local Zone. It is a subset of the VPC CIDR block.
- You can create multiple VPCs in the same Region or in different Regions, and in the same account or different accounts.
- Follow best practices when you design your VPC.

Module 6: Creating a Networking Environment

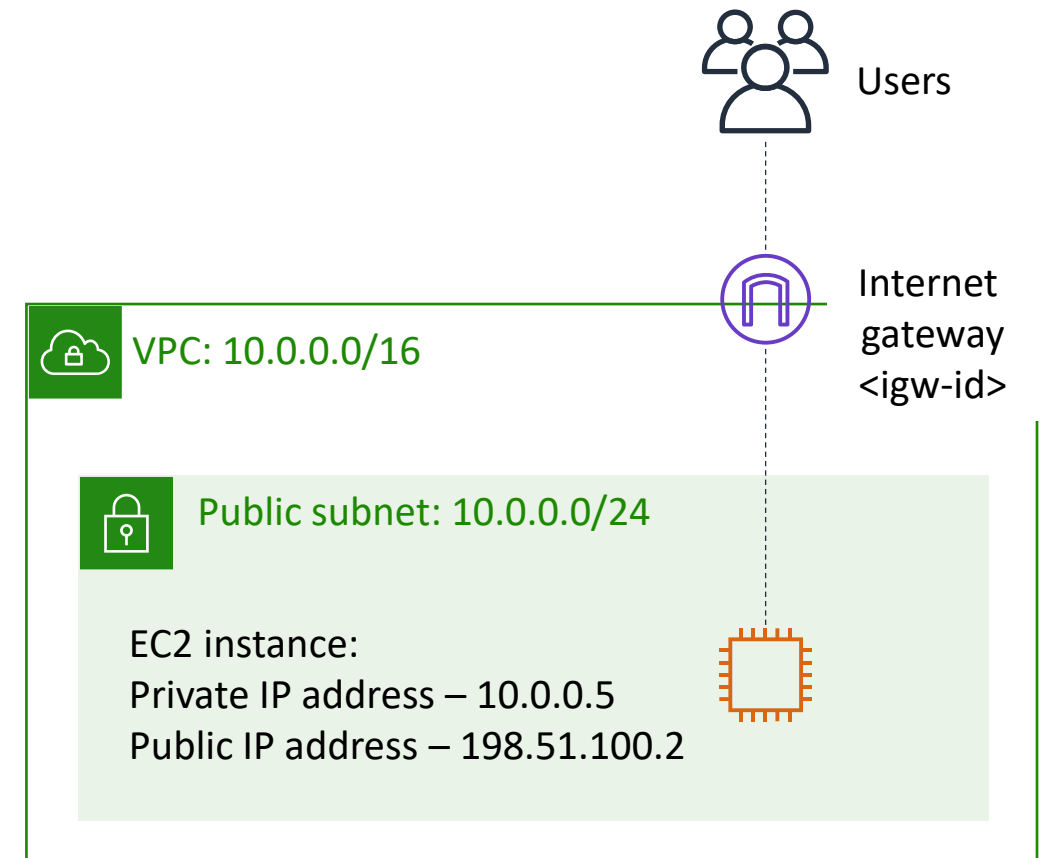
Section 3: Connecting your AWS networking environment to the internet

Creating a public subnet



Internet gateways

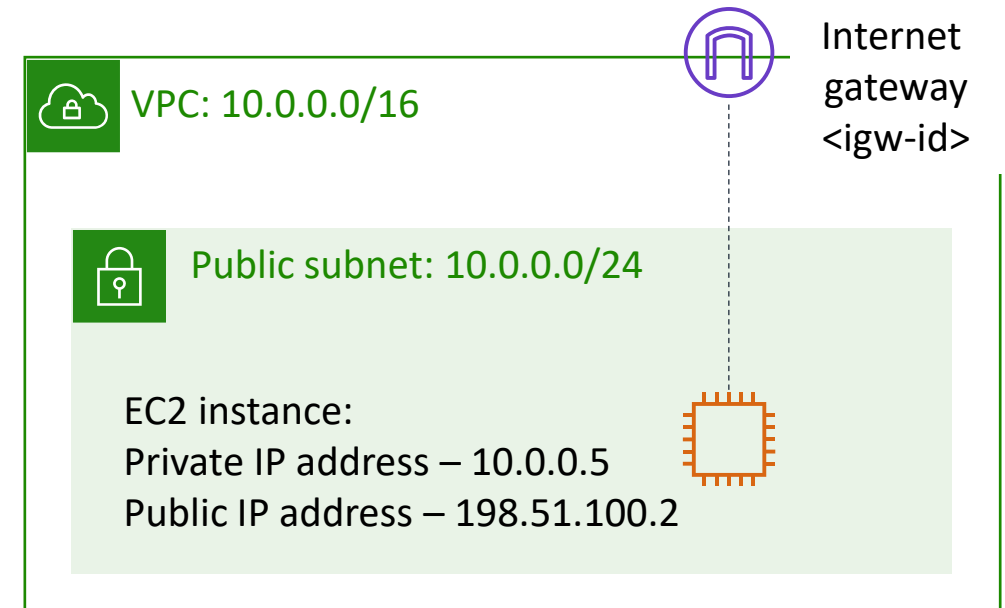
- Allow communication between resources in your VPC and the internet
- Are horizontally scaled, redundant, and highly available by default
- Provide a target in your subnet route tables for internet-routable traffic



Directing traffic between VPC resources

- **Route tables** are required to direct traffic between VPC resources
- Each VPC has a **main (default)** route table
- All subnets **must** be associated with a route table
- You can create **custom** route tables

Best practice: Use custom route tables for each subnet.



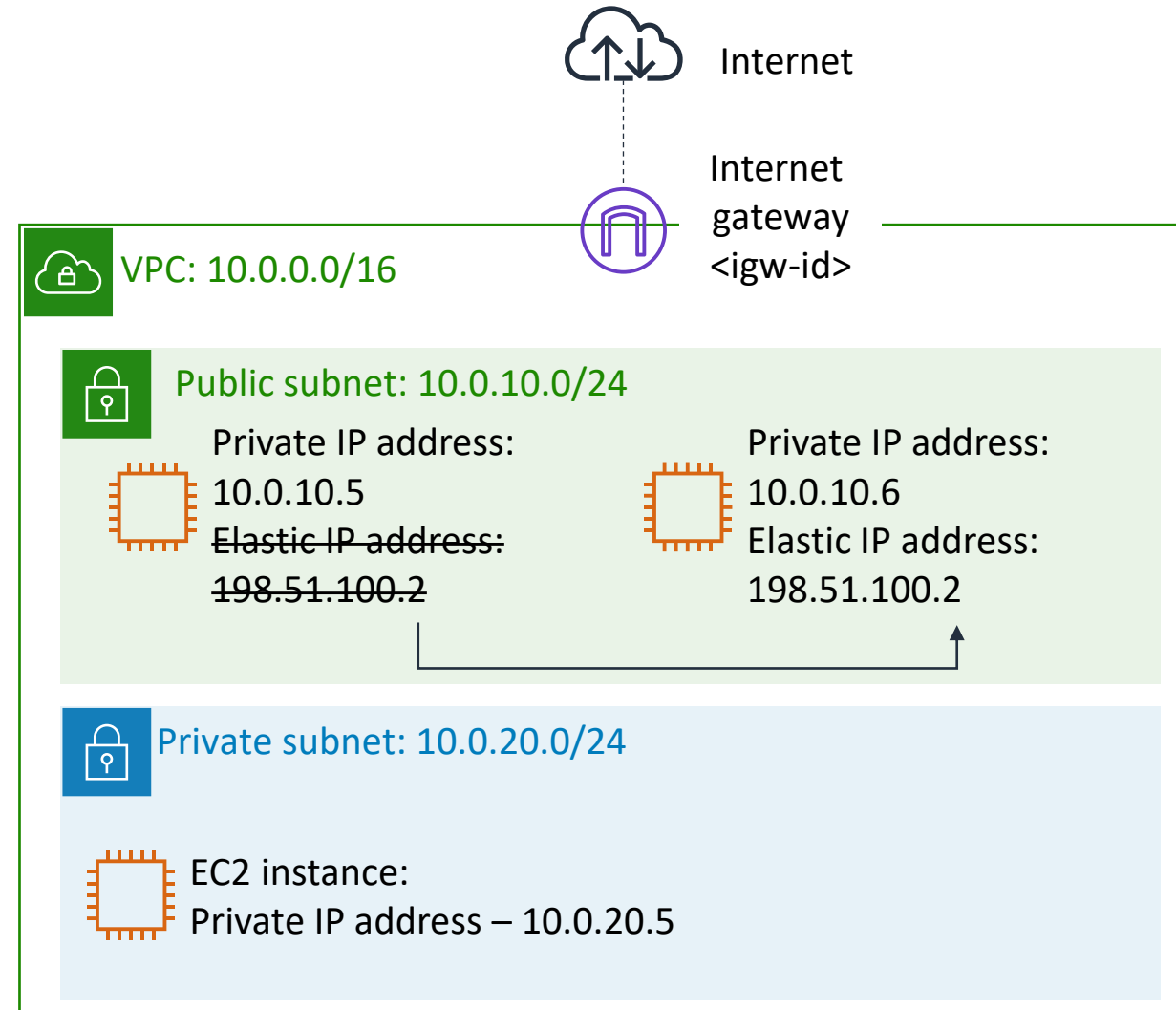
Public route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

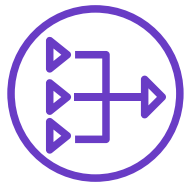
Remapping an IP address from one instance to another

🔗 Elastic IP addresses

- Are static, public IPv4 addresses associated with your AWS account
- Can be associated with an instance or elastic network interface
- Can be remapped to another instance in your account
- Are useful for redundancy when load balancers are not an option



Connecting private subnets to the internet



NAT gateways

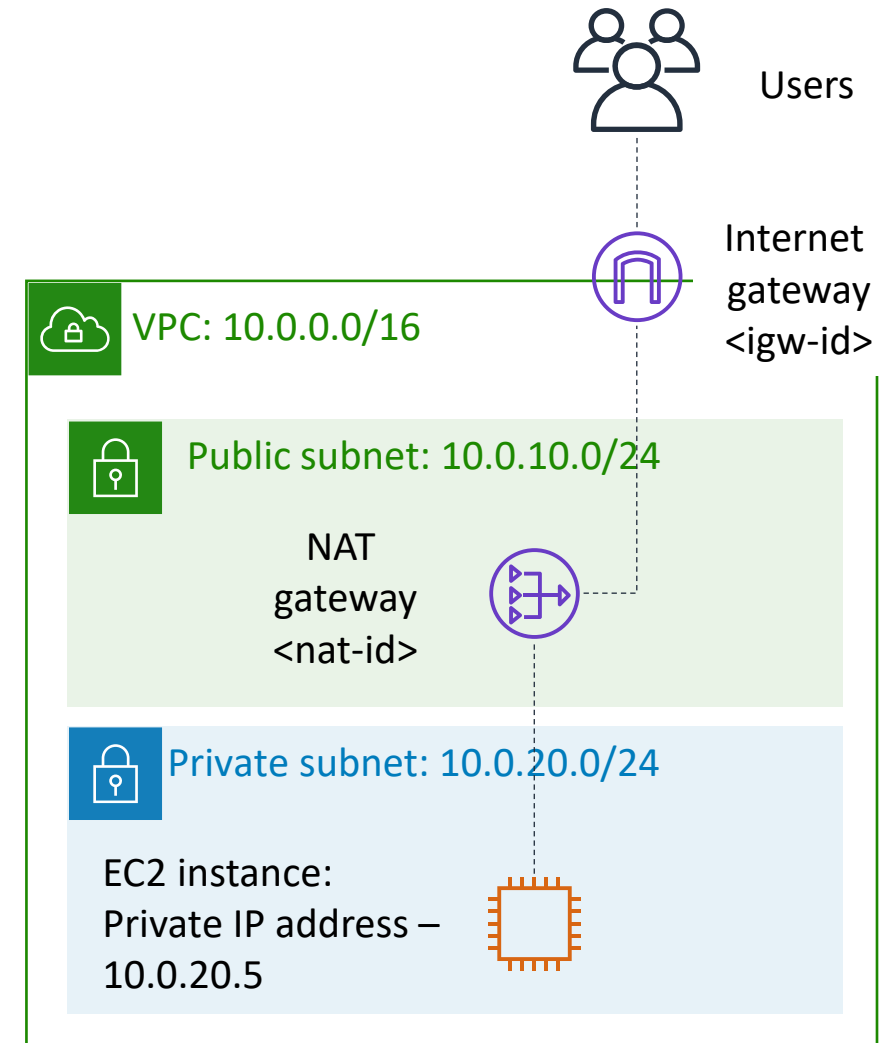
- Enable instances in a private subnet to initiate outbound traffic to the internet or other AWS services
- Prevent private instances from receiving inbound connection requests from the internet

Public route table

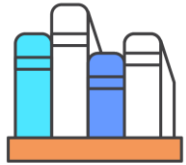
Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Private route table

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<nat-id>



Subnet use case examples (1 of 2)



Data store instances



Batch-processing instances

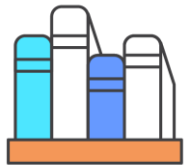


Backend instances



Web application instances

Subnet use case examples (2 of 2)



Data store instances



Private subnet



Batch-processing instances



Private subnet



Backend instances



Private subnet



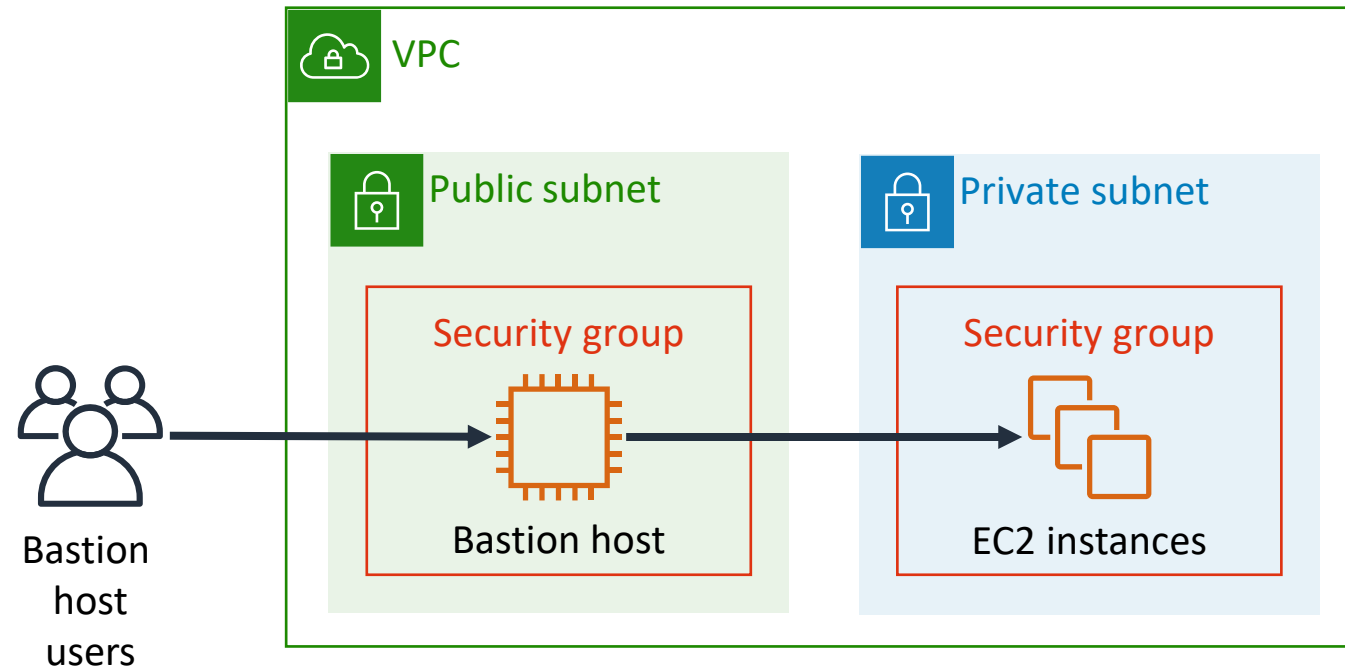
Web application instances



Public or private subnet

Bastion hosts

- A server whose purpose is to provide access to a private network from an external network
- Must minimize the chances of penetration



Demonstration: Creating a VPC



Section 3 key takeaways



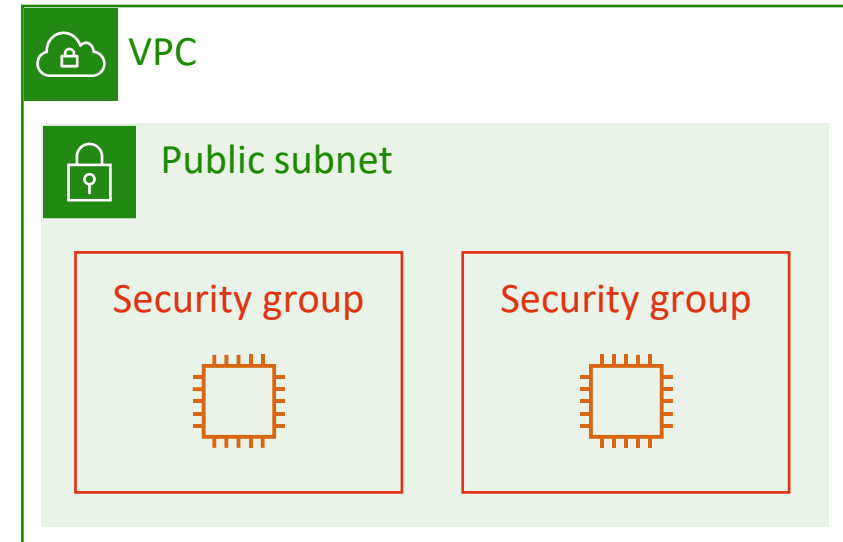
- An **internet gateway** allows communication between instances in your VPC and the internet.
- **Route tables** control traffic from your subnet or gateway.
- **Elastic IP addresses** are static, public IPv4 addresses that can be associated with an instance or elastic network interface. They can be remapped to another instance in your account.
- **NAT gateways** enable instances in the private subnet to initiate outbound traffic to the internet or other AWS services.
- A **bastion host** is a server whose purpose is to provide access to a private network from an external network, such as the internet.

Module 6: Creating a Networking Environment

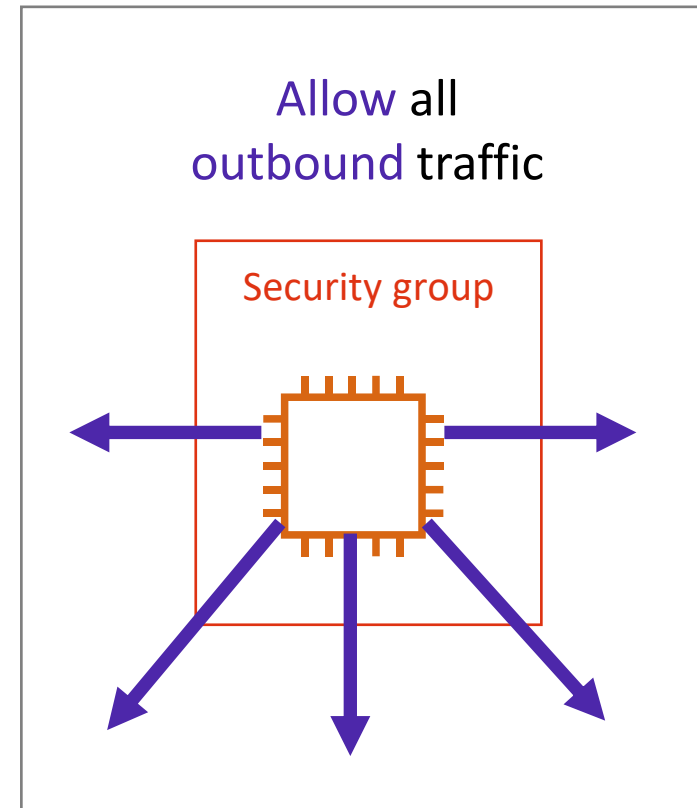
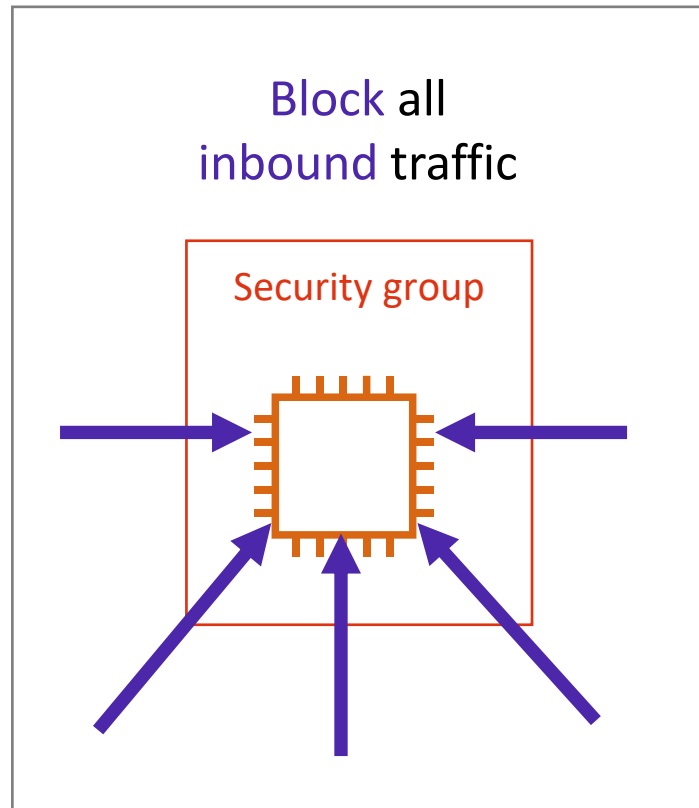
Section 4: Securing your AWS networking environment

Security groups

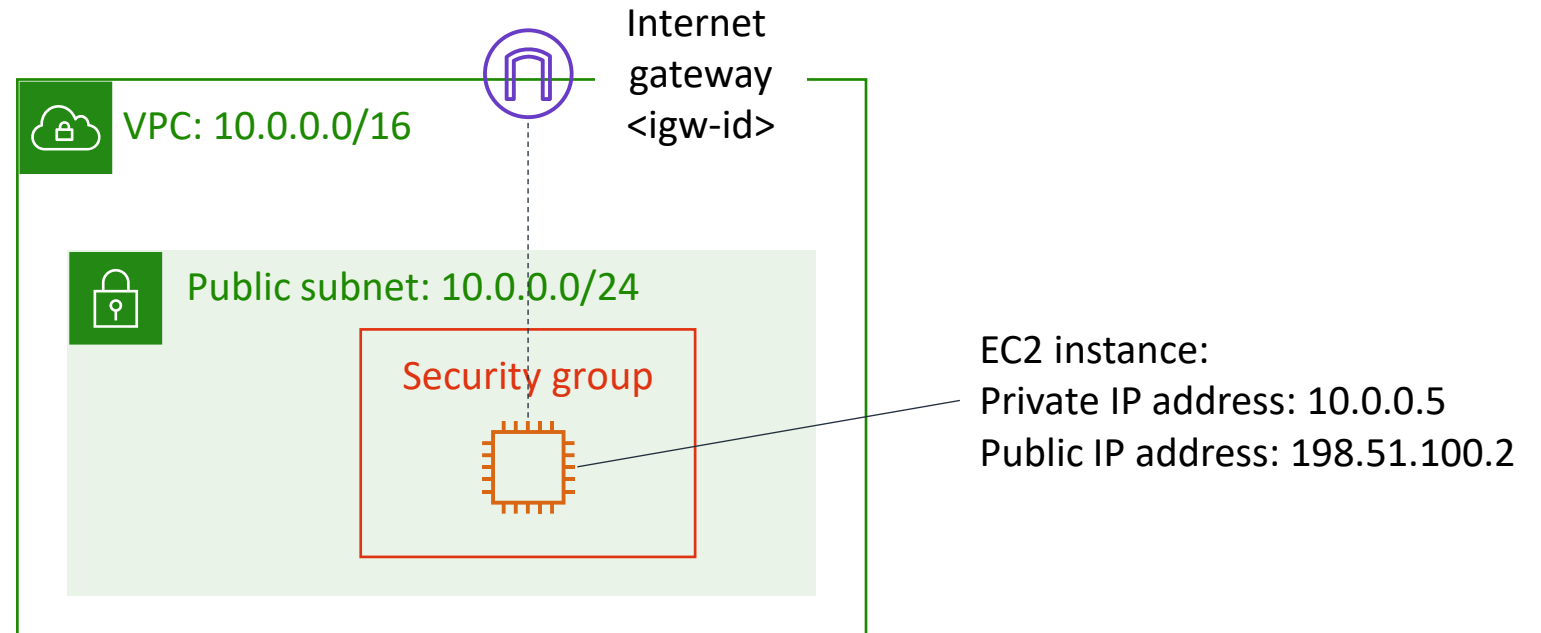
- Are **stateful firewalls** that control inbound and outbound traffic to AWS resources
- Act at the **level of the instance or network interface**



Default security groups

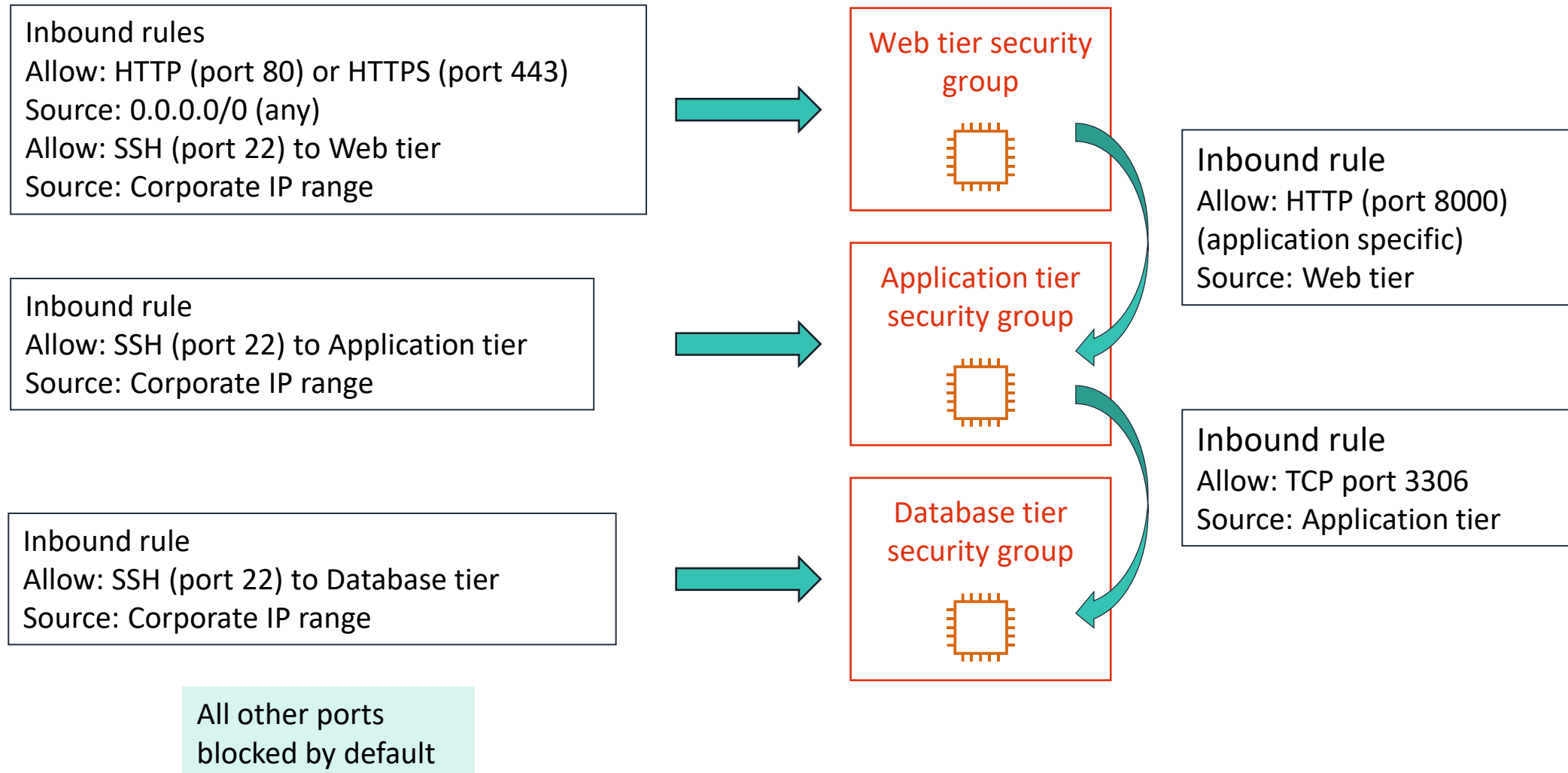


Custom security groups



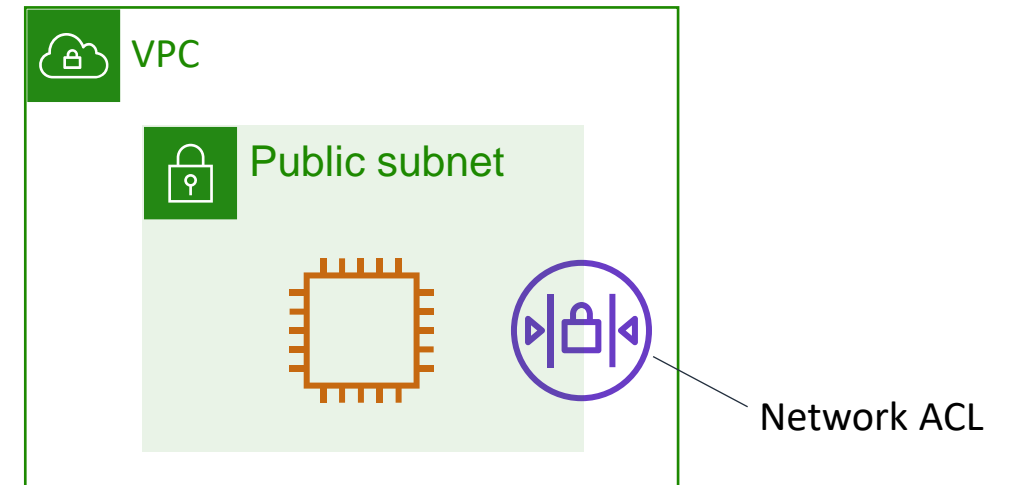
Inbound				
Type	Protocol	Port Range	Source	Destination
HTTP	TCP	80	Anywhere	Allow web access

Chaining security groups

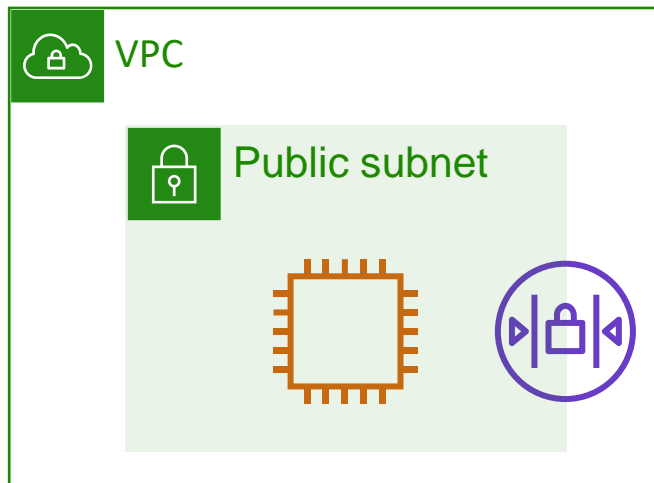


Network access control lists (network ACLs)

- Act at the **subnet level**
- **Allow all inbound and outbound traffic** by default
- Are **stateless firewalls** that require explicit rules for both inbound and outbound traffic



Recommended for
specific network security requirements only



Nacl-11223344

Inbound:

Rules # 100: SSH 172.31.1.2/32 **ALLOW**

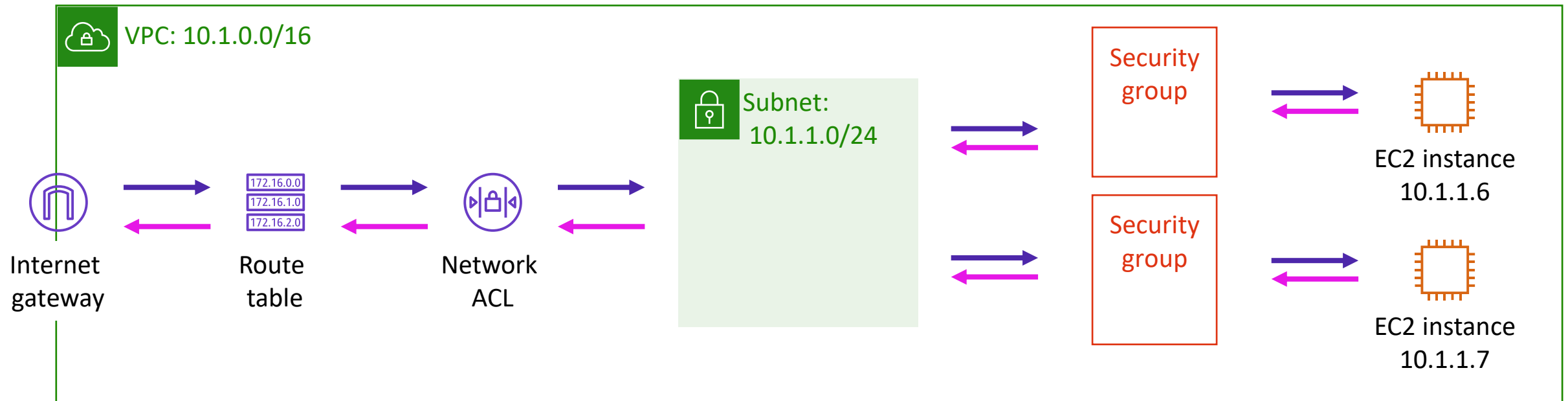
Rules # *: ALL traffic 0.0.0.0/0 **DENY**

Outbound:

Rules # 100: Custom TCP 172.31.1.2/31 **ALLOW**

Rules # *: All traffic 0.0.0.0/0 **DENY**

Structure your infrastructure with multiple layers of defense



Review: How to create a public subnet

To create a **public subnet** to allow communication between instances in your VPC and the internet, you must:



Attach an **internet gateway** to your VPC.

Destination	Target
10.0.0.0/16	local
0.0.0.0/0	<igw-id>

Point your instance subnet's **route table** to the internet gateway.



Make sure that your instances have **public IP** or **Elastic IP** addresses.



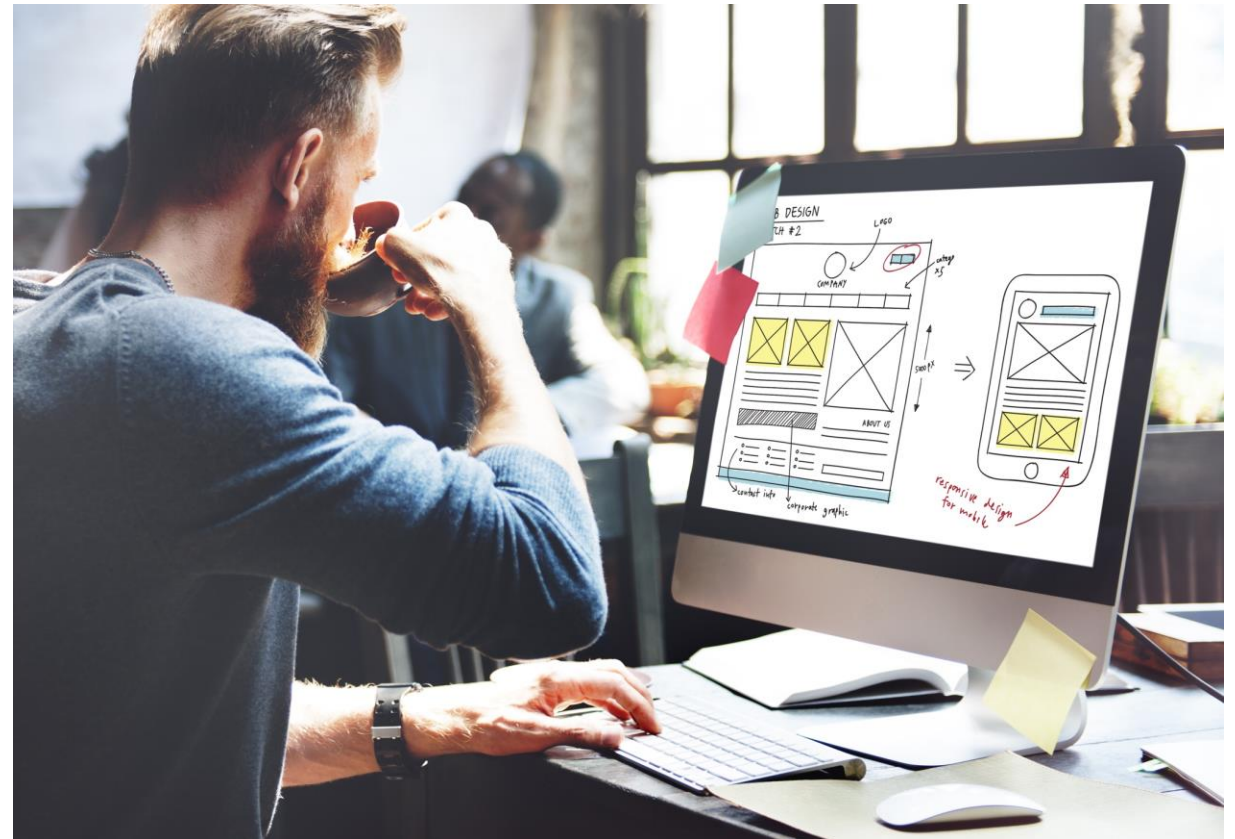
Make sure that your **security groups** and **network ACLs** allow relevant traffic to flow.

Section 4 key takeaways



- Security groups are **stateful** firewalls that act at the **instance level**
- Network ACLs are **stateless** firewalls that act at the **subnet level**
- When you set inbound and outbound rules to allow traffic to flow from the top tier to the bottom tier of your architecture, you can **chain security groups together** to isolate a security breach
- You should structure your infrastructure with **multiple layers of defense**

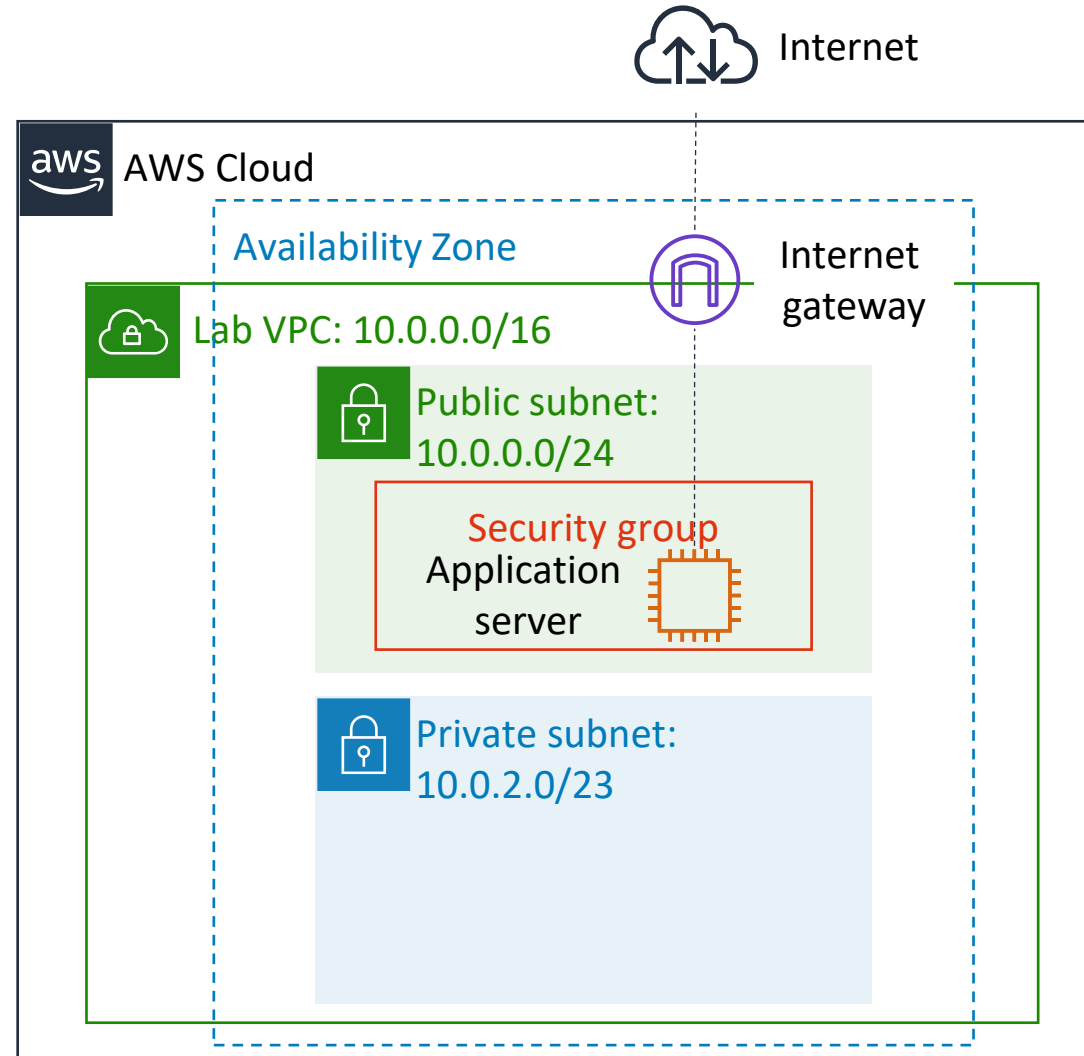
Module 6 - Guided Lab: Creating a Virtual Private Cloud



Use Amazon VPC to manually create a VPC with:

- Public and private subnets
- An internet gateway
- A route table with a route to direct internet-bound traffic to the internet gateway
- A security group for EC2 instances in the public subnet
- An application server to test the VPC

Guided lab: Final product

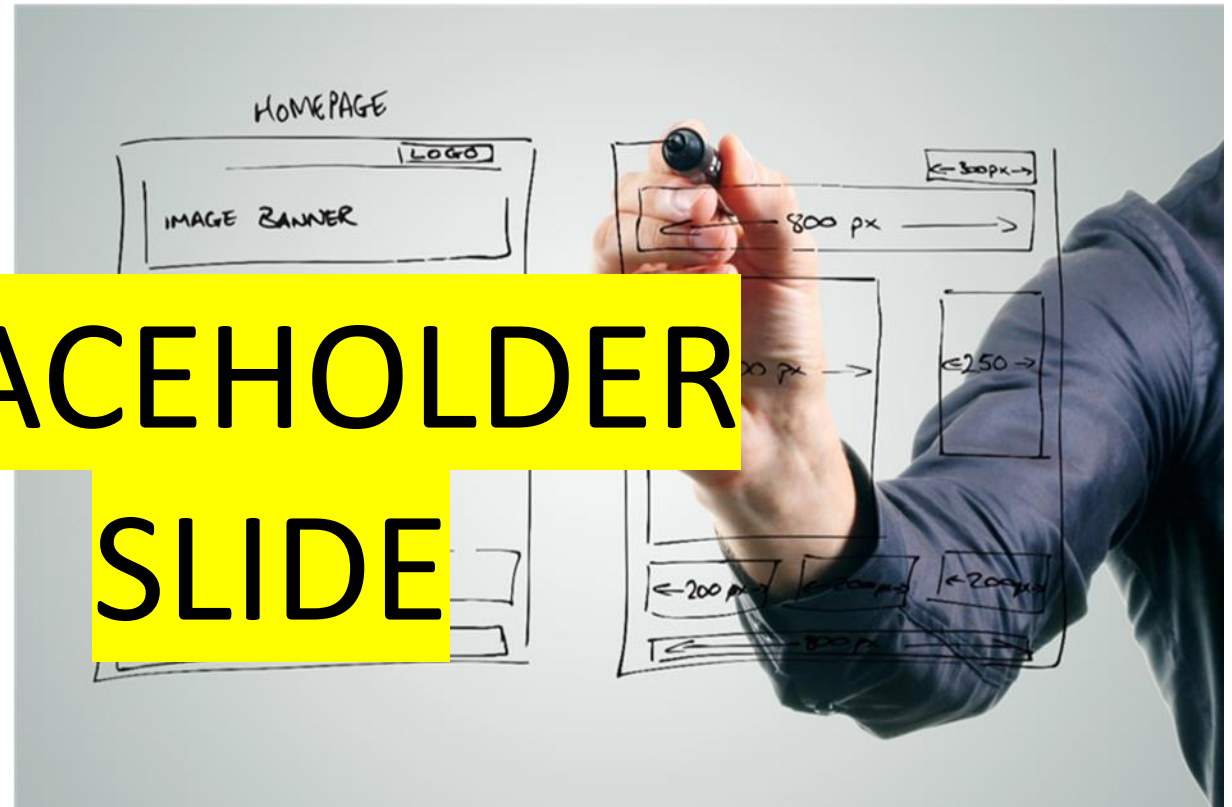


Guided lab debrief: Key takeaways



Module 6 – Challenge Lab: <Exact Lab Title>

PLACEHOLDER
SLIDE



The business need: <summary>

<tie the service or technical concept into whatever example use case the challenge labs revolve around>

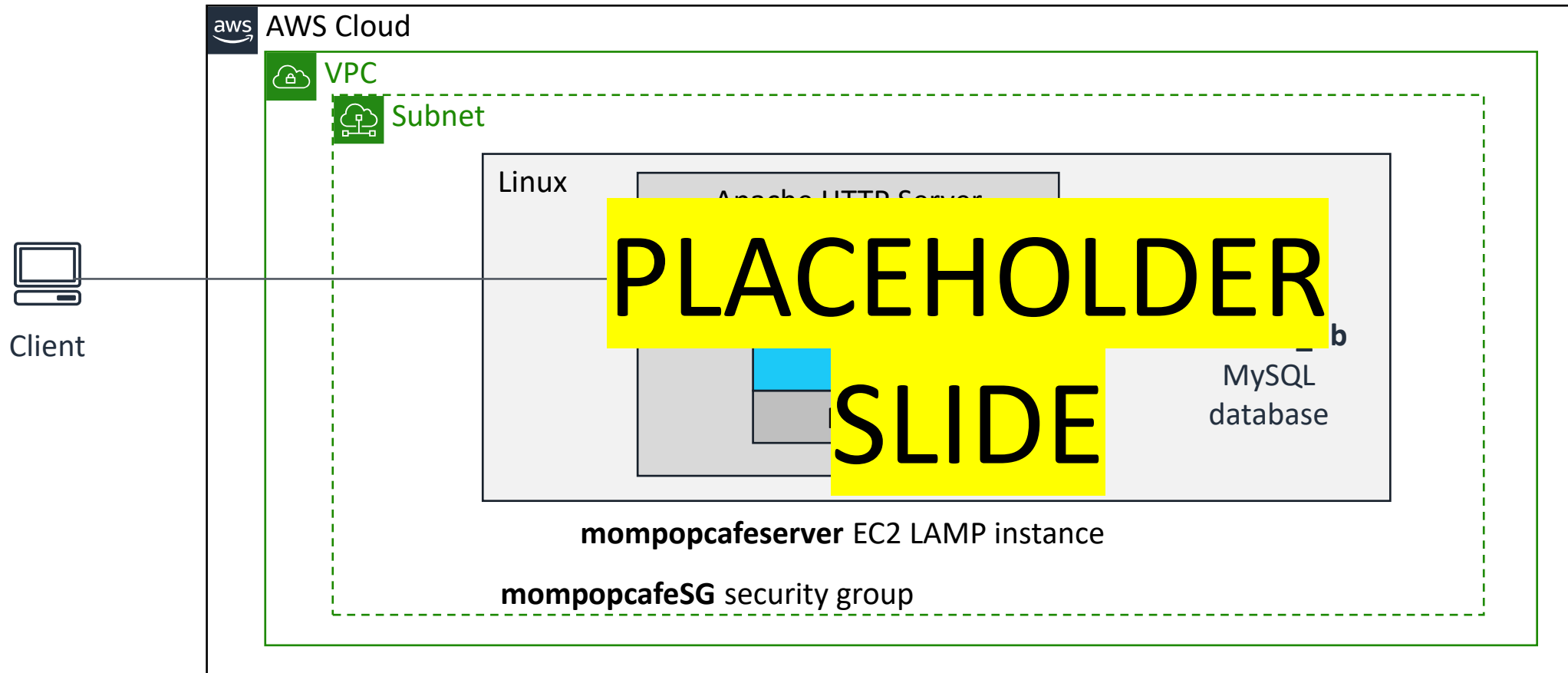
**PLACEHOLDER
SLIDE**

Challenge lab: Customer requirements

- <customer requirement #>

**PLACEHOLDER
SLIDE**

Challenge lab: Final product





PLACEHOLDER

SLIDE

Rule 6 –
Challenge Lab: <Exact
Title>

Challenge lab debrief: Key takeaways



Module 6: Creating a Networking Environment

Module wrap-up

In summary, in this module, you learned how to:

- Explain the foundational role of VPC in AWS Cloud networking
- Identify how to connect your AWS networking environment to the internet
- Describe how to isolate resources within your AWS networking environment
- Create a VPC with subnets, an internet gateway, route tables, and a security group

Complete the knowledge check



Sample exam question

You have an application running on multiple Amazon Elastic Compute Cloud (Amazon EC2) instances in a single Availability Zone. The application calls a third-party application programming interface (API) via the internet.

How can you provide the third-party API with a single IP address to add to an access safelist?

- A. Assign an Elastic IP address to the instances.
- B. Assign a public IP address to the instances.
- C. Put the instances behind a NAT gateway.
- D. Put the instances behind a Network Load Balancer.

- [VPCs and Subnets](#)
- [One to Many: Evolving VPC Design](#)
- [AWS Single VPC Design](#)
- [AWS re:Invent 2018: Your Virtual Data Center: VPC Fundamentals and Connectivity Options](#)
- [AWS Networking Fundamentals](#)

Thank you

© 2020 Amazon Web Services, Inc. or its affiliates. All rights reserved. This work may not be reproduced or redistributed, in whole or in part, without prior written permission from Amazon Web Services, Inc. Commercial copying, lending, or selling is prohibited. Corrections or feedback on the course, please email us at: aws-course-feedback@amazon.com. For all other questions, contact us at: <https://aws.amazon.com/contact-us/aws-training/>. All trademarks are the property of their owners.

