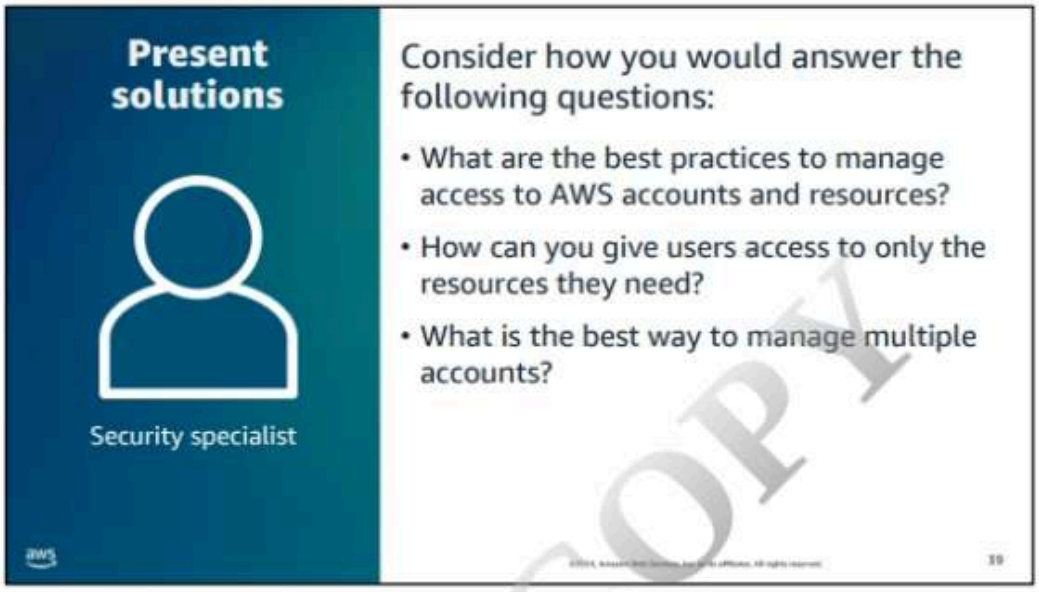
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Imagine that you are now ready to talk to the security specialist and present solutions that meet their architectural needs. Think about how you would answer the questions from the beginning of the lesson about account security.

Your answers should include the following solutions:

- Create IAM users, user groups, and roles to manage access to AWS accounts and resources.
- Build security policies with allow and deny statements. Use permissions boundaries as a protective layer.
- Use SCPs in AWS Organizations to manage multiple accounts.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Module review

In this module, you learned about the following:

- ✓ Principals and identities
- ✓ Security policies
- ✓ Managing multiple accounts

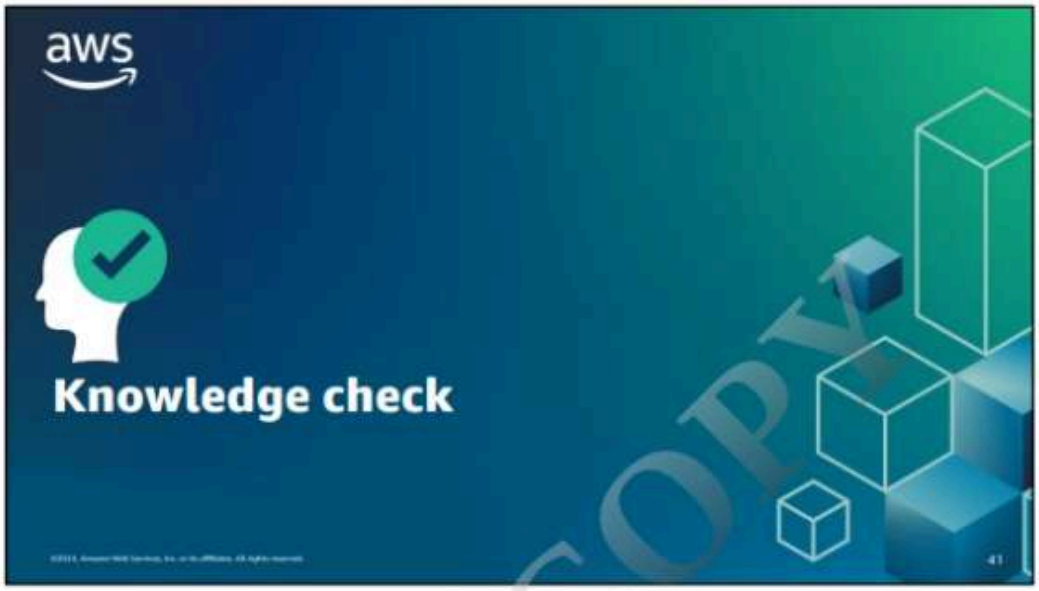
Next, you review the following:

Knowledge check

40

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 1

Which option can be attached to a user, group, or role?

Choice	Response
A	Resource-based policies
B	AWS Security Token Service (AWS STS)
C	Security groups
D	Identity-based policies

42

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 1 and answer

Which option can be attached to a user, group, or role?

Choice	Response
A	Resource-based policies
B	AWS Security Token Service (AWS STS)
C	Security groups
D correct	Identity-based policies

The correct answer is D, identity-based policies.

Attach managed and inline policies to IAM identities. These identities include users, user groups, and roles.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 2

Which option sets permissions on a specific resource and requires a principal to be listed in the policy?

Choice	Response
A	Identity-based policies
B	Service control policies (SCPs)
C	Resource-based policies
D	Permissions boundaries

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 2 and answer

Which option sets permissions on a specific resource and requires a principal to be listed in the policy?

Choice	Response
A	Identity-based policies
B	Service control policies (SCPs)
C correct	Resource-based policies
D	Permissions boundaries

45

The correct answer is C, resource-based policies.

Resource-based policies attach inline policies to resources. The most common examples of resource-based policies are Amazon S3 bucket policies and IAM role trust policies.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 3

Which options are elements of an IAM user's long-term programmatic access? (Select TWO.)

Choice	Response
A	Username
B	Access key ID
C	Password
D	Secret access key
E	Multi-factor authentication (MFA) token

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 3 and answer

Which options are elements of an IAM user's long-term programmatic access? (Select TWO.)

Choice	Response
A	Username
B correct	Access key ID
C	Password
D correct	Secret access key
E	Multi-factor authentication (MFA) token

The correct answers are B and D, access key ID and secret access key.

An IAM user's programmatic access does not use a username and password authentication. Instead, they use a unique access key ID and secret access key. MFA tokens cannot be used with long-term credentials (IAM user access keys and root user access keys).

For IAM users to use MFA they must call the AWS Security Token Service (STS) with an MFA token to obtain temporary credentials. For more information see "Requesting credentials for users in untrusted environments" in the *AWS Identity and Access Management User Guide* at https://docs.aws.amazon.com/IAM/latest/UserGuide/id_credentials_temp_request.html#api_getsessiontoken.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

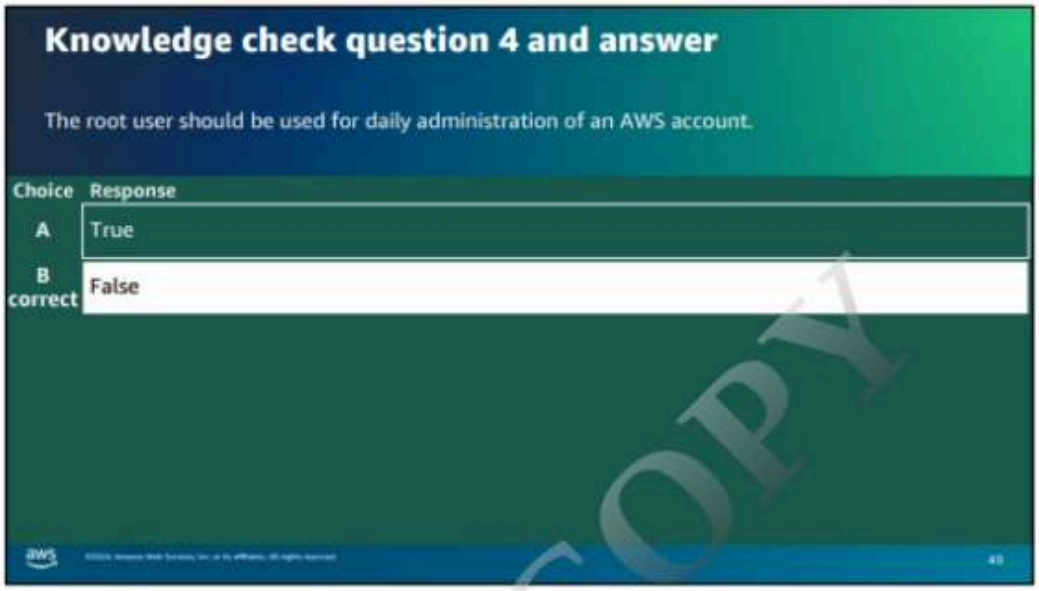
Knowledge check question 4

True or False: The root user should be used for daily administration of an AWS account.

Choice	Response
A	True
B	False

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



The correct answer is B, false.

You do not use the root user for your everyday tasks, even the administrative ones. Instead, adhere to the best practice of using the root user only to create your first IAM user. Then, securely lock away the root user credentials and use them to perform only a few account and service management tasks.

For more information about the root user, see "AWS account root user" in the *AWS Identity and Access Management User Guide* at https://docs.aws.amazon.com/IAM/latest/UserGuide/id_root-user.html.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 5

Which option can only be managed with AWS Organizations?

Choice	Response
A	Service control policies (SCPs)
B	Resource-based policies
C	Permissions boundaries
D	Identity-based policies

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 5 and answer

Which option can only be managed with AWS Organizations?

Choice	Response
A correct	Service control policies (SCPs)
B	Resource-based policies
C	Permissions boundaries
D	Identity-based policies

The correct answer is A, service control policies (SCPs).

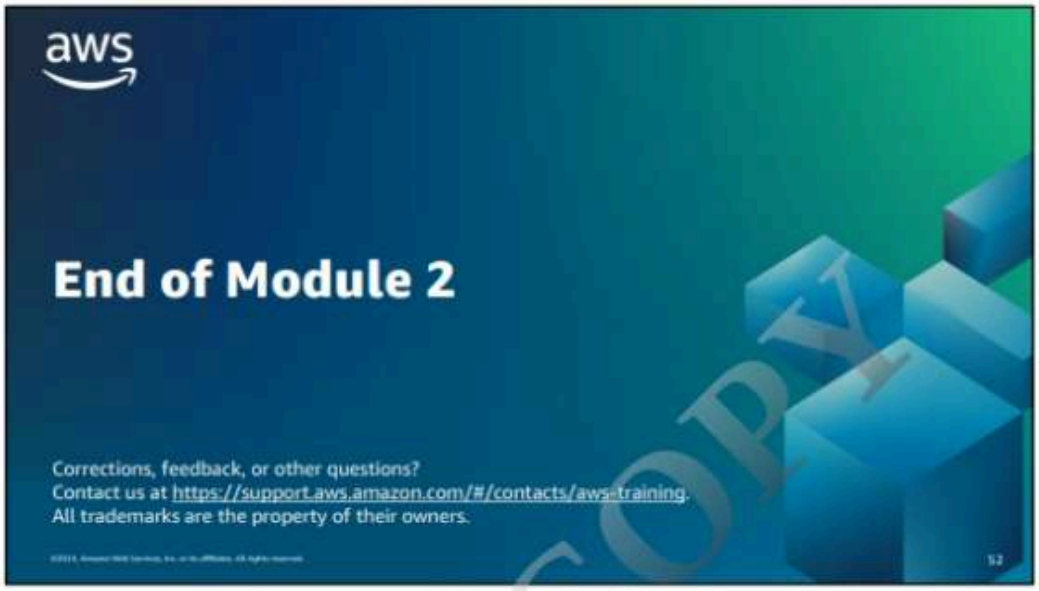
SCPs offer central control over the maximum available permissions for all accounts in your organization. You manage your SCPs through AWS Organizations. You can attach an SCP to the organization root, to an organizational unit (OU), or directly to an account.

The administrator must still attach identity-based policies or resource-based policies to IAM users or roles, or to the resources in your accounts to actually grant permissions. The effective permissions are the logical intersection between what the SCP allows and what the identity-based policies and resource-based policies allow.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Poll question

Which network components are you familiar with? Choose all that apply:

A. IP addressing and subnetting

B. Switching and routing

C. Network security

D. None of the above

2

DO NOT COPY
javierartiga.espublico@gmail.com

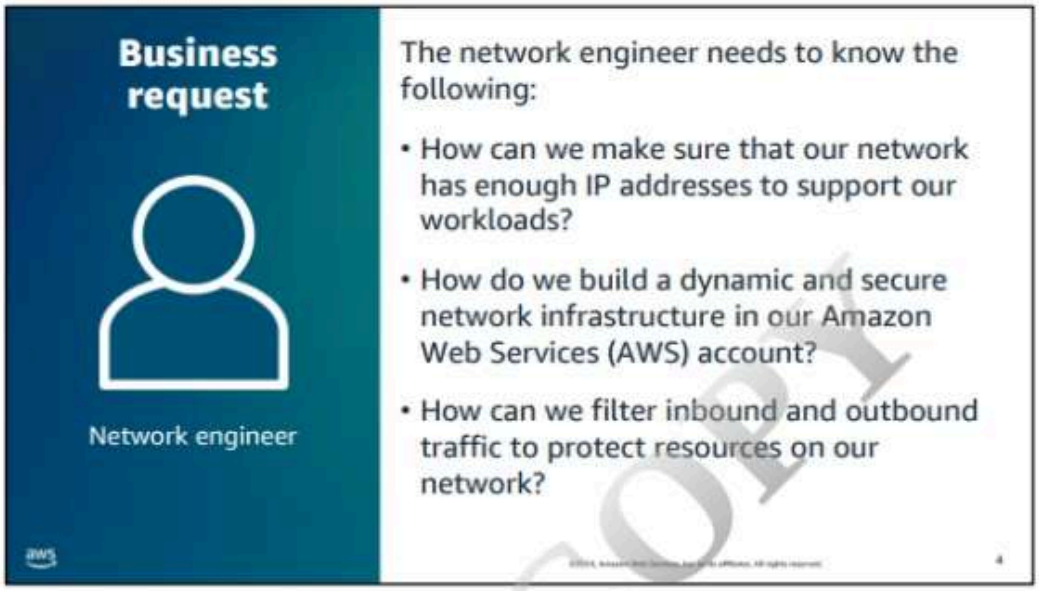
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Module overview

- Business request
- IP addressing
- Amazon Virtual Private Cloud (VPC) fundamentals
- VPC traffic security
- Present solutions
- Capstone check-in
- Knowledge check

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

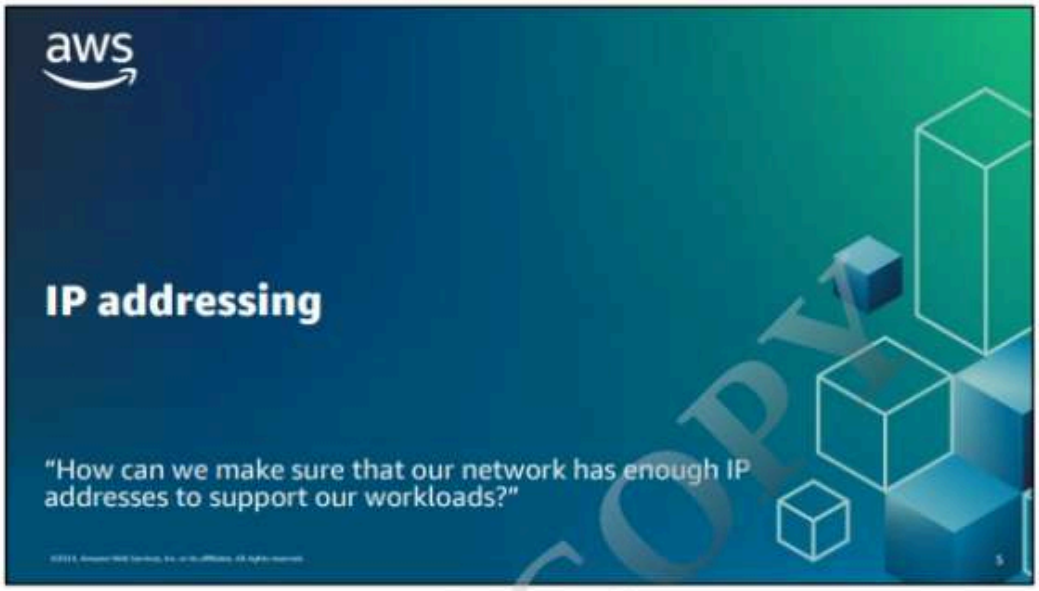


Imagine that your network engineer meets with you to discuss how to build networking infrastructure in the cloud. Here are some questions that they are asking.

At the end of this module, you meet with the network engineer and present some solutions.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

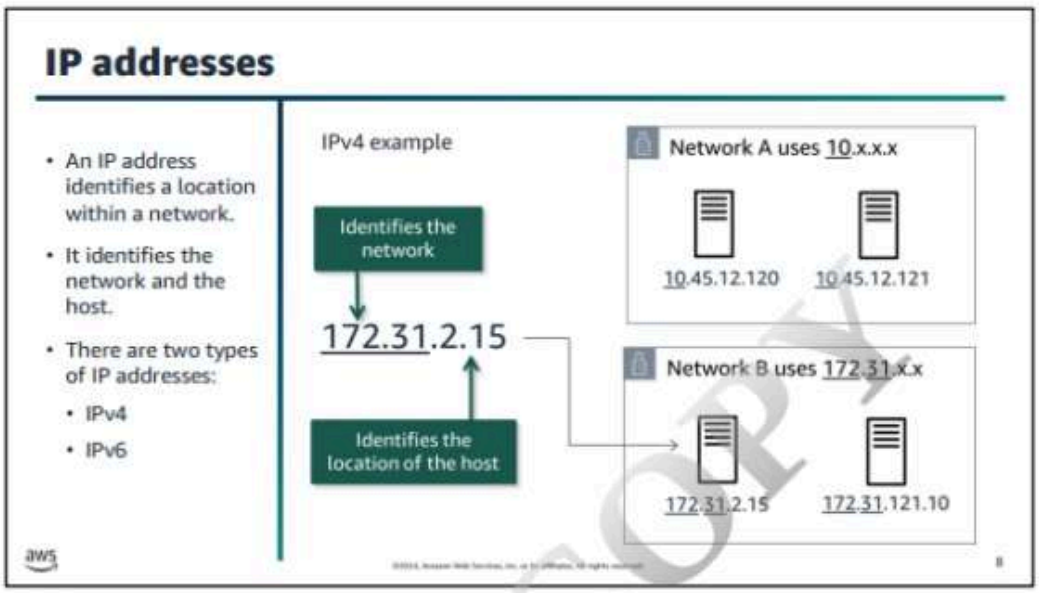


The network engineer asks, "How can we make sure that our network has enough IP addresses to support our workloads?" The engineer and their team must start planning to build a network. They would like you to explain how to define IP address ranges on AWS.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



An IP address includes information about the location of a resource within a network. One part of the address identifies the network, and another part identifies the host.

This example shows a sample IPv4 address of 172.31.2.15. Each of the four numbers separated by dots is called an octet. The destination network, network B, uses the first two octets to identify the network and the last two to identify the host. However, network A uses only the first octet to identify the network and the other three to identify the host.

There are two IP address types:

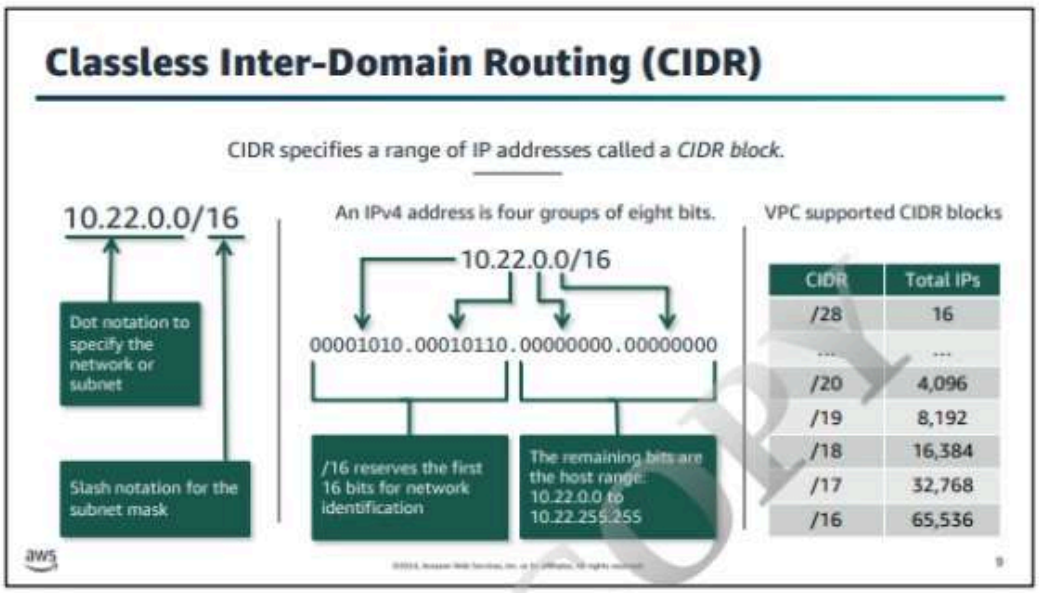
- **IPv4 addresses**
  IPv4 was developed in the early 1980s and uses 32-bit addresses. The bits are grouped into four sets of eight bits, called octets. Addresses in IPv4 are written by using numeric dot-decimal notation. IPv4 addresses can create **4.3 billion addresses.**
- **IPv6 addresses**
  IPv6 was developed in 1998 to replace IPv4. IPv6 uses 128-bit addresses. IPv6 addresses are eight groups of four hexadecimal digits, for a total of 16 octets. The groups are written with a colon as a separator. A full IPv6 address is often expressed in a shortened form. For example, 50b2:6400:0000:0000:0000:6c3a:b17d:0:10a9 can be written as 50b2:6400::6c3a:b17d:0:10a9. The addresses have a capacity for **340 trillion trillion trillion addresses**. IPv6 supports automatic configuration. Global unicast address (GUA) is a unique IPv6 address that is assigned to a host interface.

Note: You cannot remove IPv4 support from your VPC. However, it's possible to create IPv6 only subnets.

[Image Description: Two networks: A and B contains two hosts each. Network A uses the IP range 10.x.x.x and has two hosts with the IP addresses 10.45.12.120 and 10.45.12.121. Network B uses the IP range 172.31.x.x and has two hosts with the IP addresses 172.31.2.15 and 172.31.121.10. The IP address for the first host of network B with an IP address of 172.31.2.15 is split in two, where 172.31 identifies the network and 2.15 identifies the location of the host. **End Description.]**

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



CIDR notation example with 10.22.0.0/16, showing network and host bit allocation. Includes a table of VPC-supported CIDR blocks and IP ranges.

CIDR notation defines an IP address range for a network or a subnet. This range is referred to as a CIDR block. When you use VPC components to build your network in AWS, you specify CIDR blocks for your VPC and subnets. You must allocate enough IP addresses to support the resources on your network. Your VPC can have up to five CIDR blocks, and their address ranges cannot overlap.

A CIDR block identifies the network by using dot notation. It specifies the subnet mask by using slash notation. The subnet mask defines which portion of the IP address is dedicated to network identification and which can be used for host IP addresses. For example, an IPv4 address has 32 bits divided into four octets. A subnet mask of /16 reserves the first 16 bits, or two octets, for network identification. The remaining 16 bits are used for host identification.

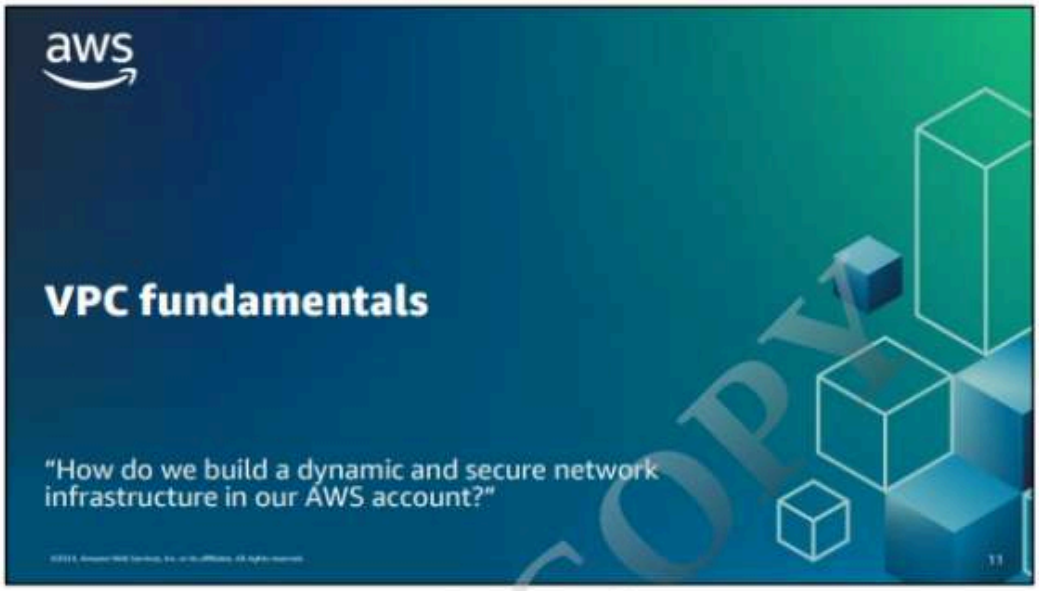
Each octet can have a number between 0 and 255, which creates a range of 65,536 addresses. The network reserves some addresses, which are not usable: one address for the network and four addresses for other reasons that will be discussed later.

Amazon Virtual Private Cloud (Amazon VPC) supports IPv4 and IPv6 addressing, and it has different CIDR block size limits for each. By default, all VPCs must have IPv4 CIDR blocks—you can't change this behavior. You can optionally associate an IPv6 CIDR block with a dual-stack VPC.

You can assign block sizes between /28 (16 IP addresses) and /16 (65,536 IP addresses) for IPv4 subnets. The size of the IPv6 CIDR block for IPv6-only subnets has a fixed prefix length of /64.

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

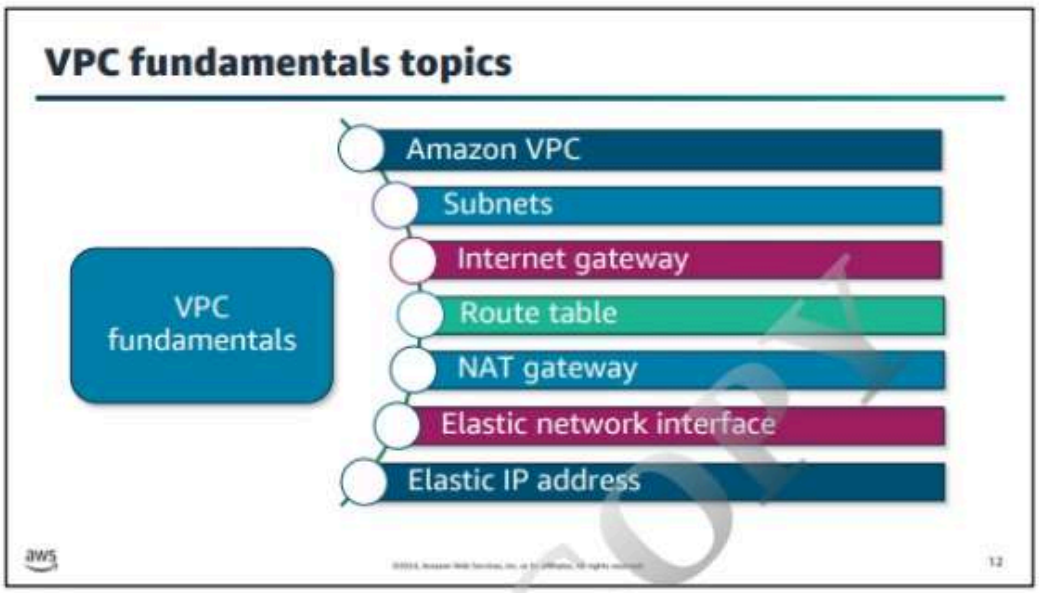


The network engineer asks, "How do we build a dynamic and secure network infrastructure in our AWS account?" The network engineer needs to identify the elements that build an elastic, secure virtual network that includes private, public, and protected subnets. They would like you to explain the common networking components on AWS and how they work together.

DO NOT COPY
javierartiga.espublico@gmail.com

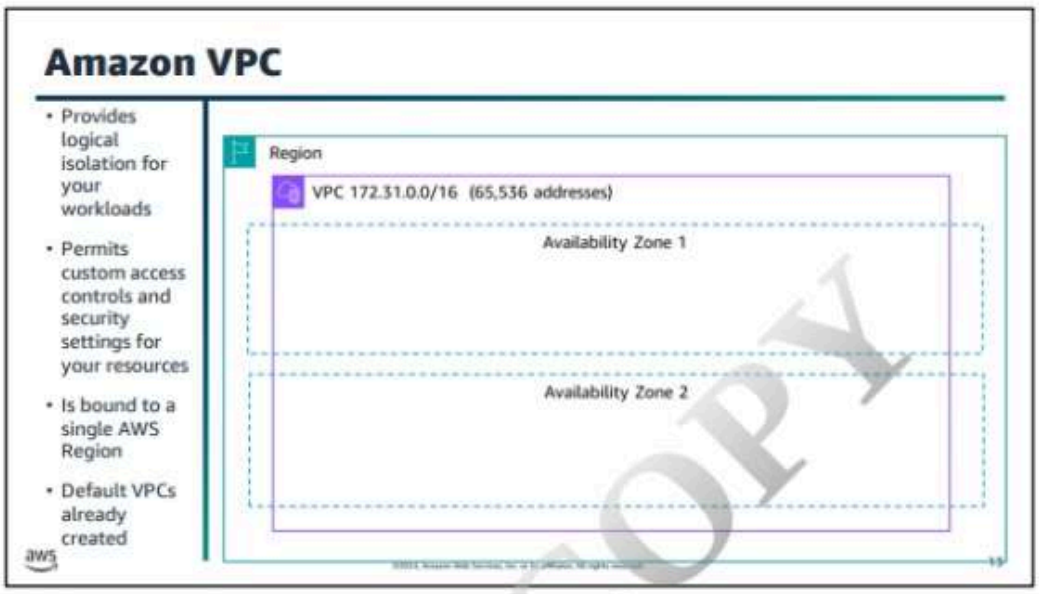
© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Amazon Virtual Private Cloud (Amazon VPC) is your network environment in the cloud. With Amazon VPC, you can launch AWS resources into a virtual network that you have defined. A VPC deploys into one of the AWS Regions and can host resources from any Availability Zone within its Region.

Amazon VPC is designed to provide greater control over the isolation of your environments and their resources. With Amazon VPC, you can do the following:

- Select your own IP address range.
- Create subnets.
- Configure route tables and network gateways.

This VPC uses a CIDR block of 172.31.0.0/16, which provides a range of 65,536 addresses.

You can use both IPv4 and IPv6 in your VPC for secure and easy access to resources and applications. A VPC is a virtual network that is dedicated to your AWS account.

For more information about Amazon VPC, see "What is Amazon VPC?" in the *Amazon Virtual Private Cloud User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/what-is-amazon-vpc.html.

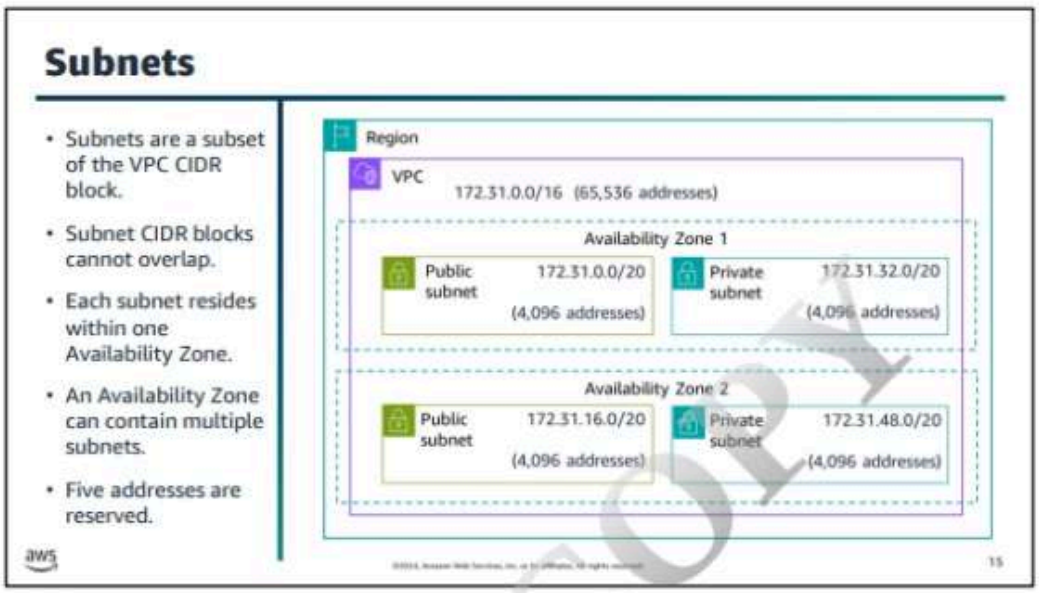
Each AWS account comes with a default Amazon VPC that is preconfigured for you to use immediately. The default Amazon VPC is suitable for getting started quickly and for launching public instances, such as a blog or simple website. However, it is recommended to remove the default VPC.

For more information about default VPCs, see "Default VPCs" in the *Amazon Virtual Private Cloud User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/default-vpc.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



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. A subnet resides within one Availability Zone.

In this example, the VPC CIDR block 172.31.0.0/16 allows a total of 65,536 IP addresses. These addresses are divided among its four subnets as follows:

- Public subnet 1: 172.31.0.0/20 allows 4,096 addresses in the range 172.31.0.1 – 172.31.15.254.
- Public subnet 2: 172.31.16.0/20 allows 4,096 addresses in the range 172.31.16.1 – 172.31.31.254.
- Private subnet 1: 172.31.32.0/20 allows 4,096 addresses in the range 172.31.32.1 – 172.31.47.254.
- Private subnet 2: 172.31.48.0/20 allows 4,096 addresses in the range 172.31.48.1 – 172.31.63.254.

These subnets do not use all of the available addresses in the VPC. You can use these extra addresses to support future growth.

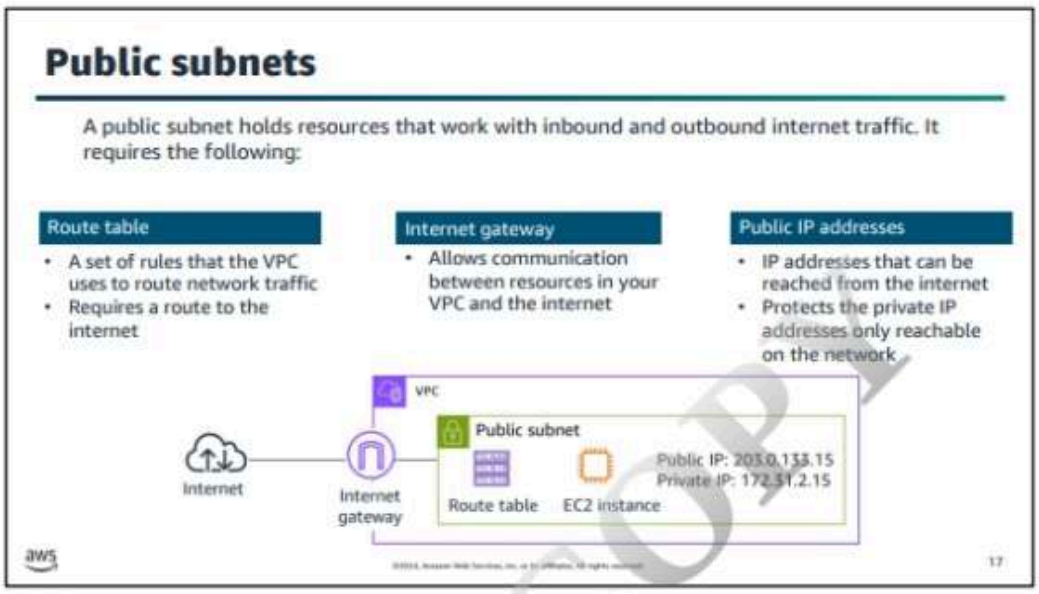
The first four IP addresses and the last IP address in each subnet CIDR block are not available and cannot be assigned to an instance. For example, in a subnet with CIDR block 10.0.0.0/24, the following five IP addresses are reserved:

- **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. AWS does not support broadcast in a VPC. Therefore, it reserves this address.

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.

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Your public subnet configuration acts as a two-way door, which allows traffic to flow in either direction, invited or not invited. Because it has no automatic outbound routing, you must configure a subnet to be public.

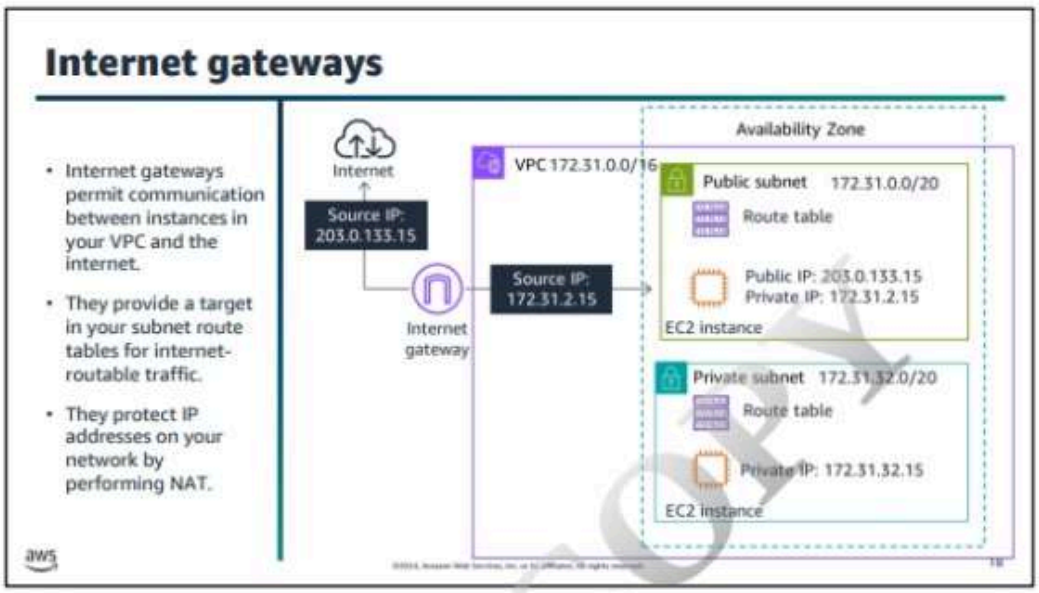
A public subnet requires the following:

- Internet gateway: The internet gateway allows communication between resources in your VPC and the internet.
- Route table: A route table contains a set of rules (routes) that are used to determine where network traffic is directed. It can direct traffic to the internet gateway.
- Public IP addresses: These addresses are accessible from the internet. Public IP addresses obscure the private IP addresses, which are only reachable within the network.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



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 supports IPv4 and IPv6 traffic.

An internet gateway serves two purposes:
1. It provides a target in your route table for internet-routable traffic.
2. It protects IP addresses on your network by performing NAT.

Provides a target in your route table for internet-routable traffic
A subnet does not allow outbound traffic by default. Your VPC uses route tables to determine where to route traffic. To allow your VPC to route internet traffic, you create an outbound route in your route table with an internet gateway as a target, or destination.

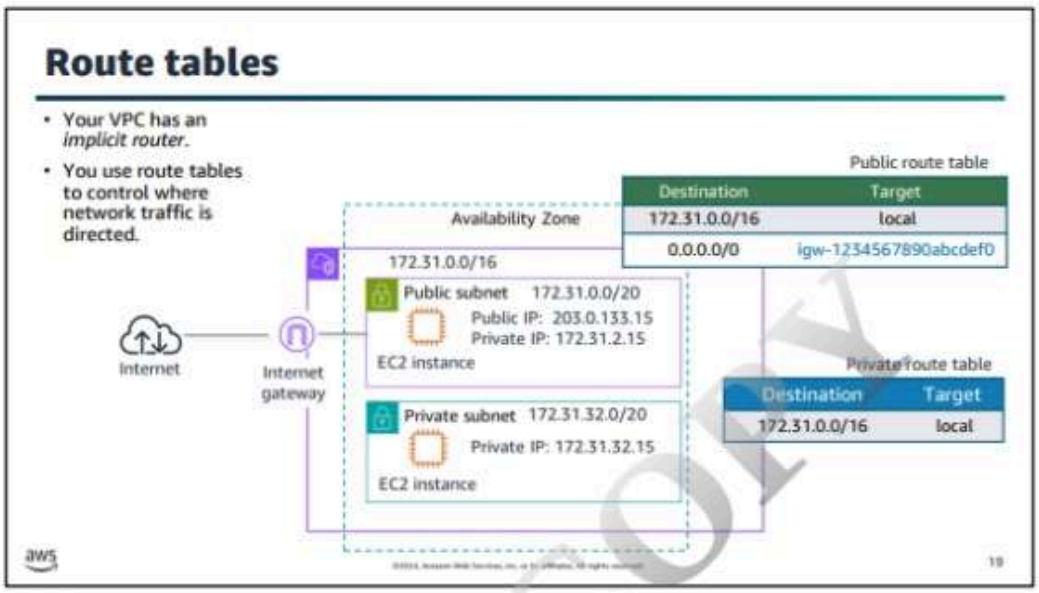
Protects IP addresses on your network by performing NAT
Resources on your network that connect to the internet should use two kinds of IP addresses:
- **Private IP**: Use private IPs for communication within your private network. These addresses are not reachable over the internet.
- **Public IP**: Use public IP addresses for communication between resources in your VPC and the internet. A public IP address is reachable over the internet.

An internet gateway performs NAT by mapping a public and private IP address. In this example, the internet gateway translates the source IP of a request from the private IP used on the network (172.31.2.15) to the public IP address (203.0.133.15). The recipient directs its response to the public IP address. The internet gateway receives the response and translates the public IP to the matching private IP address. The VPC routes the response to the requester.

For more information, see "Enable VPC internet access using internet gateways" in the *Amazon Virtual Private Cloud User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Internet_Gateway.html.

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



A route table contains a set of rules (routes) that the VPC uses to determine where to direct network traffic. 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 local route permits communication for all the resources within the VPC. You can't modify the local route in a route table. Whenever you launch an instance in the VPC, the local route automatically covers that instance. You can create additional custom route tables for your VPC.

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. You can associate a subnet 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.

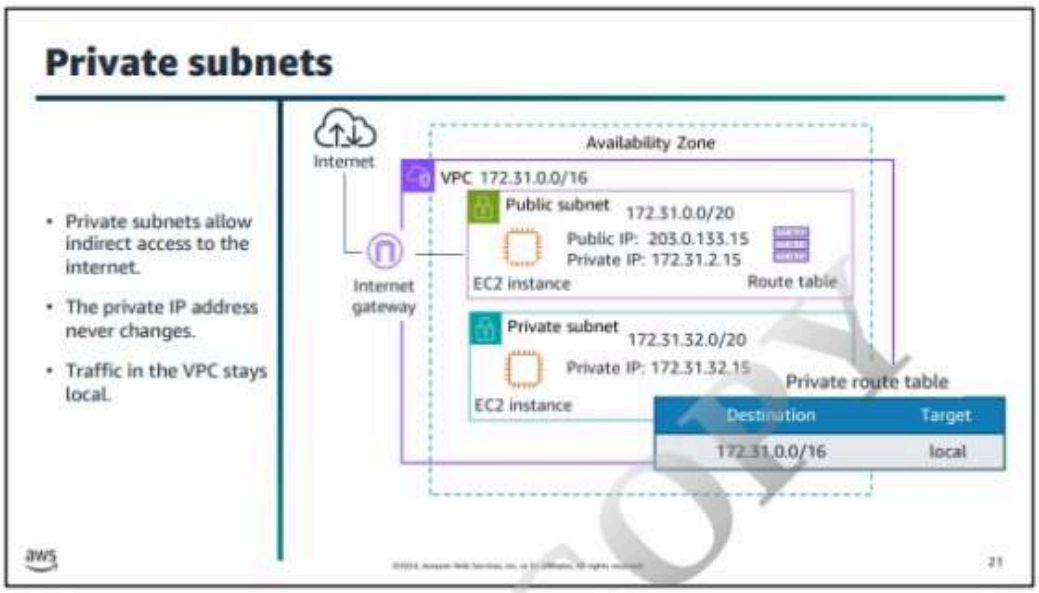
In this example, both route tables direct network traffic locally, but the public route table includes routes to the internet gateway.

For more information, see "Configure route tables" in the *Amazon Virtual Private Cloud User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Route_Tables.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



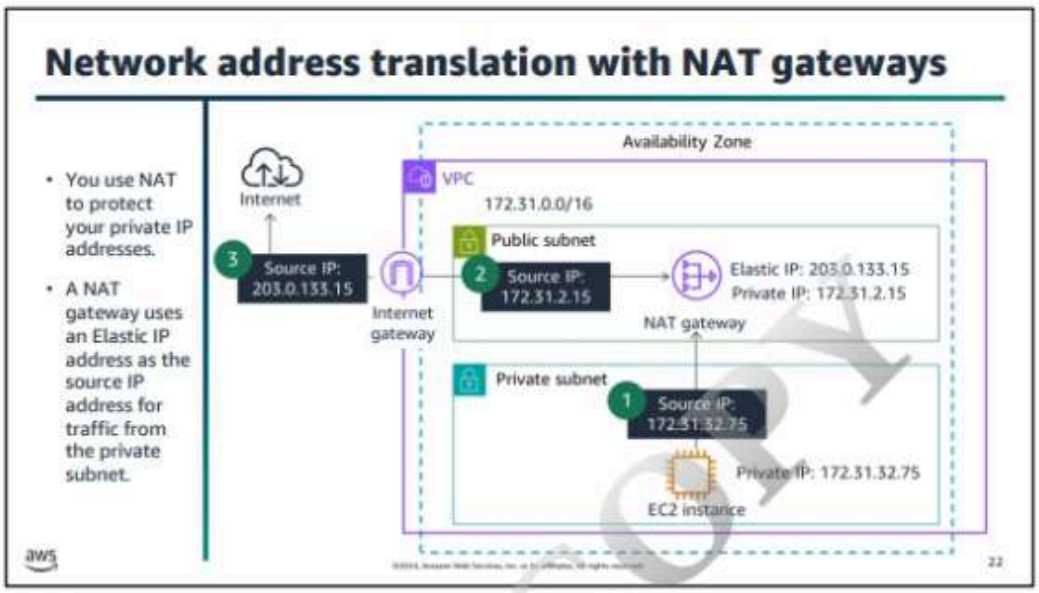
Private subnets allow indirect access to the internet. Traffic stays within your private network. A private IP address assigned to an Amazon EC2 instance will never change unless you manually assign a new IP address on the network interface of the EC2 instance.

You can put web-tier instances into a public subnet. However, we recommend that you put web-tier instances inside private subnets behind a load balancer placed in a public subnet. Elastic Load Balancing is discussed later in this course.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



NAT maps one IP address to another in a message. You use NAT for IP address conservation. Because NAT maps private IP addresses to a public IP address, you can use it to allow private IP networks to connect to the internet. A single device, such as a router, can act as an agent between the internet (public network) and a local network (private network).

NAT gateways communicate between instances in your VPC and the internet. They are horizontally scaled, redundant, and highly available by default. NAT gateways provide a target in your subnet route tables for internet-routable traffic.

- Instances in the private subnet can initiate outbound traffic to the internet or other AWS services.
- NAT gateways that AWS manages prevent private instances from receiving inbound traffic directly from the internet.

In this example, an instance in a private subnet sends a message to an external website. The destination IP address of the message remains the address of the website. However, the source IP address is translated at different points in the network.

1. The message from the private instance uses the private IP address of the instance as the source IP. You route the message to the NAT gateway by using a route table.
2. The NAT gateway changes the source IP on the message to the private IP of the NAT gateway. The NAT gateway tracks this substitution in a table. You route the message to the internet gateway by using a route table.
3. The internet gateway changes the source IP on the message to the Elastic IP address for the NAT gateway. It then forwards the packet to the public website. The internet gateway tracks this substitution in a table.
4. The external website receives the message.

Response messages use the Elastic IP address for the NAT gateway as the destination IP, and the network address translation process proceeds in reverse.

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

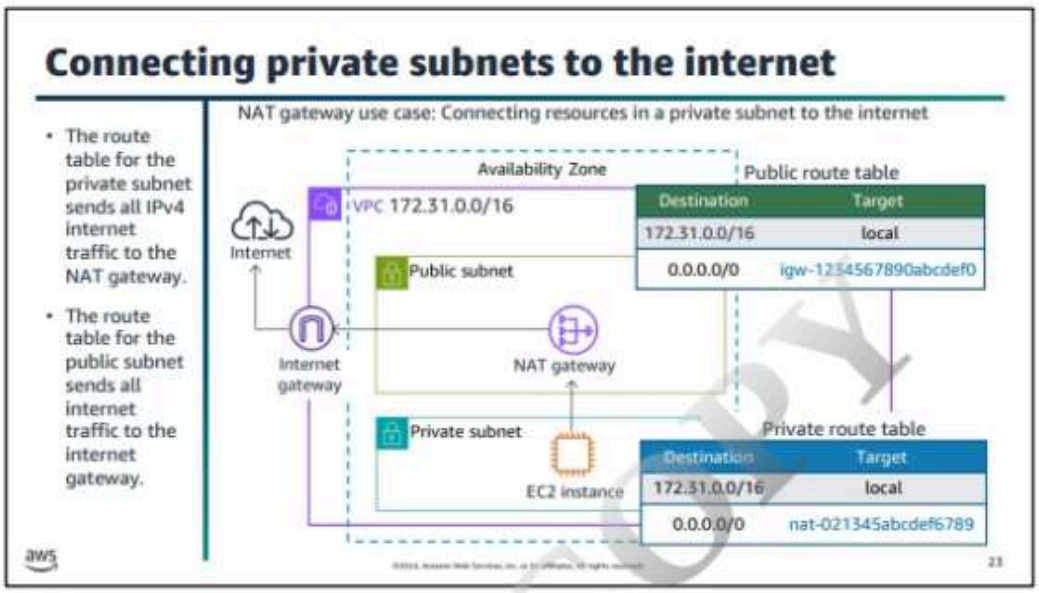
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

[Image Description: A VPC in a single Availability Zone with an internet gateway, a public subnet, and a private subnet. The VPC uses a CIDR of 172.31.0.0/16. The public subnet contains a NAT gateway with an Elastic IP address of 203.0.133.15. The private subnet contains an EC2 instance with a private IP address of 172.31.32.75. Outbound traffic to the internet from the private subnet flows to the NAT gateway. The NAT gateway translates the source address to 172.31.2.15 to obscure the private instance's private IP address and then sends the traffic to the internet gateway. The internet gateway translates the source address to 203.0.133.15. **End Description**.]

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



You can use a NAT gateway for a one-way connection between private subnet instances and the internet or other AWS services. This type of connection prevents external traffic from connecting with your private instances.

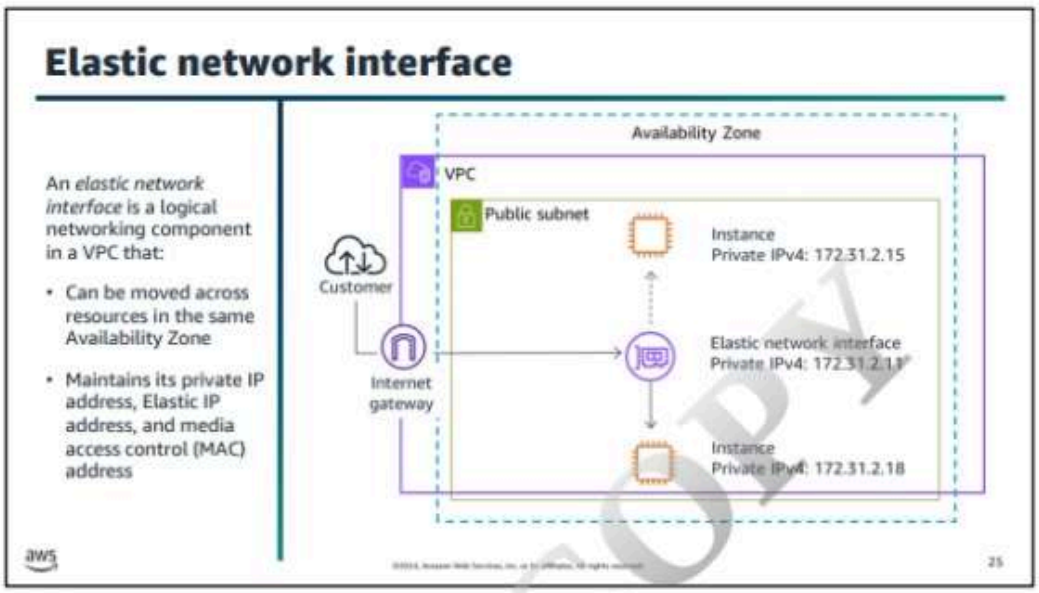
- The route table for the private subnet sends all IPv4 internet traffic to the NAT gateway.
- The NAT gateway uses its Elastic IP address as the source IP address for traffic from the private subnet.
- The route table for the public subnet sends all internet traffic to the internet gateway. This action is not supported for IPv6.

[Image Description: A VPC in a single Availability Zone with an internet gateway, a public subnet, and a private subnet. The VPC uses a CIDR of 172.31.0.0/16. The public subnet contains a NAT gateway. The private subnet contains an EC2 instance. A private route table sets the target for internet traffic in the private subnet to the NAT gateway in the public subnet. The public route table in the public subnet sets the target for internet traffic to the internet gateway. **End Description.]**

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



An *elastic network interface* is a logical networking component in a VPC that represents a virtual network card.

For Amazon EC2, you can create a network interface, attach it to an instance, detach it from an instance, and attach it to another instance. When moved to a new instance, the network interface maintains its public and Elastic IP address, private IP and Elastic IP address, and MAC address. The attributes of a network interface follow it.

When you move a network interface from one instance to another, network traffic is redirected to the new instance. Each instance in a VPC has a default network interface (the primary network interface).

Each instance is assigned a private IP address from the IP address range of your VPC. You cannot detach a primary network interface from an instance. You can create and attach additional network interfaces. Attaching multiple network interfaces to an instance is useful when you want to do the following tasks:

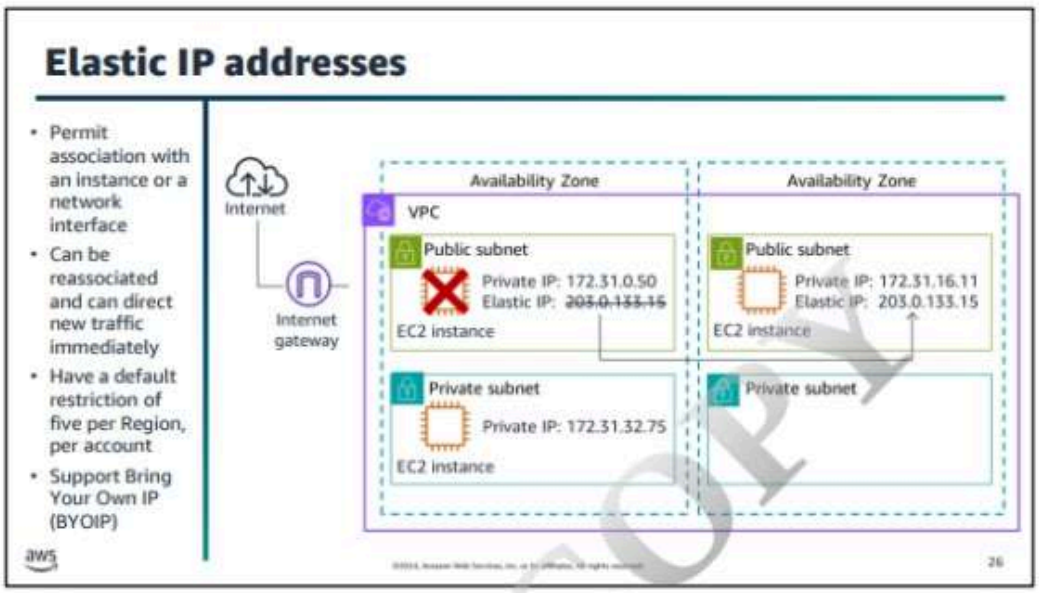
- Create a management network.
- Use network and security appliances in your VPC.
- Create dual-homed instances with workloads or roles on distinct subnets.
- Create a low-budget, high-availability solution.

You can attach a network interface in one subnet to an instance in another subnet in the same VPC. However, both the network interface and the instance must reside in the same Availability Zone. This restriction limits its use for disaster recovery (DR) scenarios, where you would want to redirect traffic to another Availability Zone.

[Image Description: A VPC in one Availability Zone with a public subnet that contains two EC2 instances and an elastic network interface. Customers connect to the VPC through an internet gateway. Traffic routes to an elastic network interface with a private IP address of 172.31.2.11. You attach the elastic network interface to an EC2 instance with a private IP address of 172.31.2.15. The interface can be reassigned to a second EC2 instance with a private IP address of 172.31.2.18. **End Description.]**

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



An Elastic IP address is a static, public IPv4 address that is designed for dynamic cloud computing. You can associate an Elastic IP address with any instance or network interface for any VPC in your account. With an Elastic IP address, you can mask the failure of an instance by rapidly remapping the address to another instance in your VPC.

In this example, an EC2 instance that uses an Elastic IP address of 203.0.133.15 fails. This Elastic IP address is assigned to a new EC2 instance in another Availability Zone.

You can move an Elastic IP address from one instance to another. The instance can be in the same VPC or another VPC. An Elastic IP address is accessed through the internet gateway of a VPC. If you set up a VPN connection between your VPC and your network, the VPN traffic traverses a virtual private gateway, not an internet gateway. Therefore, it cannot access the Elastic IP address.

You are limited to five Elastic IP addresses by default. To help conserve them, you can use a NAT device. We encourage you to use an Elastic IP address primarily to be able to remap the address to another instance in the case of instance failure. Use DNS hostnames for all other inter-node communication.

You can create Bring Your Own IP (BYOIP) addresses, but it requires significant additional configuration.

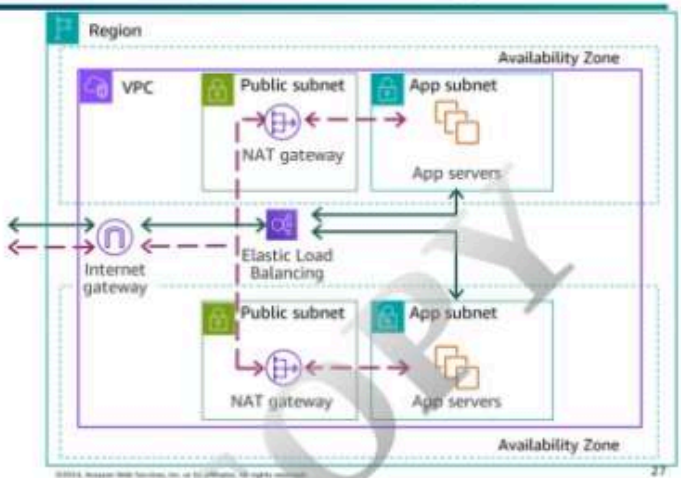
[Image Description: A VPC is in two Availability Zones with a public and private subnet in each. Customers connect to the VPC through an internet gateway. The private subnet has one EC2 instance with a private IP of 172.31.128.75. The public subnet of the first Availability Zone has one EC2 instance with a private IP of 172.31.0.50 and an associated Elastic IP of 203.0.133.15. The second Availability Zone has a public subnet with one EC2 instance with a private IP of 172.31.16.11. The EC2 instance in the public subnet of the first Availability Zone fails. Because the EC2 instance in the public subnet of the second Availability Zone is still running, the Elastic IP address is reassigned from the failed instance to the running instance. **End Description.]**

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Deploy a VPC across multiple Availability Zones

- Deploy your VPCs across multiple Availability Zones to achieve high availability.
- Create subnets in each Availability Zone.
- Deploy resources in each Availability Zone.
- Distribute traffic between the Availability Zones using load balancers.



Deploying a VPC across multiple Availability Zones creates an architecture that achieves high availability by distributing traffic while providing data security. If you have an outage in one Availability Zone, you can fail over to the other.

In this diagram of a VPC that spans two Availability Zones, the backend servers are in two private subnets in the two separate Availability Zones. They send outbound traffic to NAT gateways in public subnets located in their Availability Zone. Backend traffic from both NAT gateways routes to an internet gateway.

Elastic Load Balancing receives inbound traffic and routes it to the application servers in the private subnets of both Availability Zones.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

The network engineer asks, "How can we filter inbound and outbound traffic to protect resources on our network?" The team needs information to determine strategies for a layered security approach to VPC subnets. They want your advice for how to best control traffic in and out of your network.

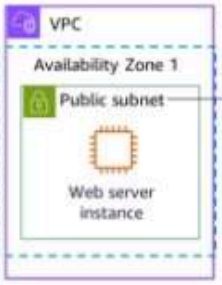
DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Network access control lists (network ACLs)

- A network ACL acts as a firewall at the subnet boundary.
- By default, it allows all inbound and outbound traffic.
- It is stateless, and requires explicit rules for all traffic.
- It evaluates rules, starting with the lowest numbered rule.



nacl-PublicSubnet

Inbound rules

Rule #	Type	Protocol	Port Range	Source	Allow or Deny
100	HTTP	TCP	80	0.0.0.0/0	*Allow*
200	HTTPS	TCP	443	0.0.0.0/0	*Allow*
*	ALL Traffic	ALL	ALL	0.0.0.0/0	Deny

Outbound rules

Rule #	Type	Protocol	Port Range	Destination	Allow or Deny
100	Custom TCP Rule	TCP	1024-65535	0.0.0.0/0	*Allow*
*	ALL Traffic	ALL	ALL	0.0.0.0/0	Deny

29

A network access control list (network 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. Every VPC automatically comes with a default network ACL. It allows all inbound and outbound IPv4 traffic. You can create a custom network ACL and associate it with a subnet. By default, custom network ACLs deny all inbound and outbound traffic until you add rules.

A network ACL contains a numbered list of rules, which are evaluated in order, starting with the lowest numbered rule. If a rule matches traffic, the rule is applied even if any higher-numbered rule contradicts it. Each network ACL has a rule whose number is an asterisk. This rule denies a packet that doesn't match any of the numbered rules.

Components of a network ACL rule include the following items:

- **Rule number** – Rules are evaluated starting with the lowest numbered rule.
- **Type** – The type of traffic, for example, Secure Shell (SSH). You can also specify all traffic or a custom range.
- **Protocol** – You can specify any protocol that has a standard protocol number.
- **Port range** – The listening port or port range for the traffic, for example, 80 for HTTP traffic.
- **Source** – For inbound rules only, the source of the traffic (CIDR range).
- **Destination** – For outbound rules only, the destination for the traffic (CIDR range).
- **Allow or Deny** – Whether to allow or deny the specified traffic.

Network ACLs are stateless, which means that responses to allowed inbound traffic are subject to the rules for outbound traffic (and the other way around).

For more information about network ACLs, see "Control subnet traffic with network access control lists" in the *Amazon Virtual Private Cloud User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html.

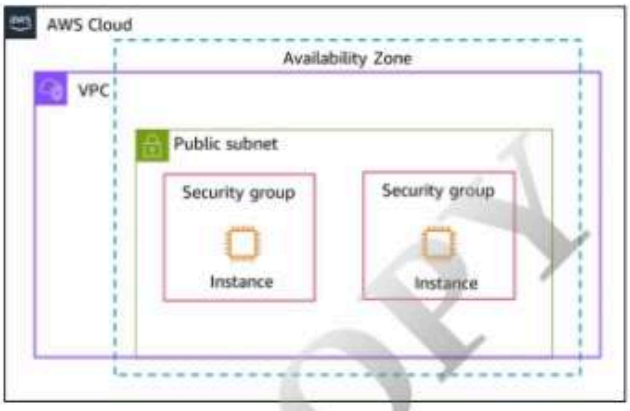
DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Security groups

- A security group is a virtual firewall that controls inbound and outbound traffic into AWS resources.
- It allows traffic based on IP protocol, port, or IP address.
- It uses stateful rules.



A *security group* acts as a virtual firewall for your instance to control inbound and outbound traffic. Security groups act at the network interface level, not the subnet level, and they support Allow rules only. The default group allows inbound communication from other members of the same group and outbound communication to any destination. Traffic can be restricted by any IP protocol, by service port, and by source or destination IP address (individual IP address or CIDR block).

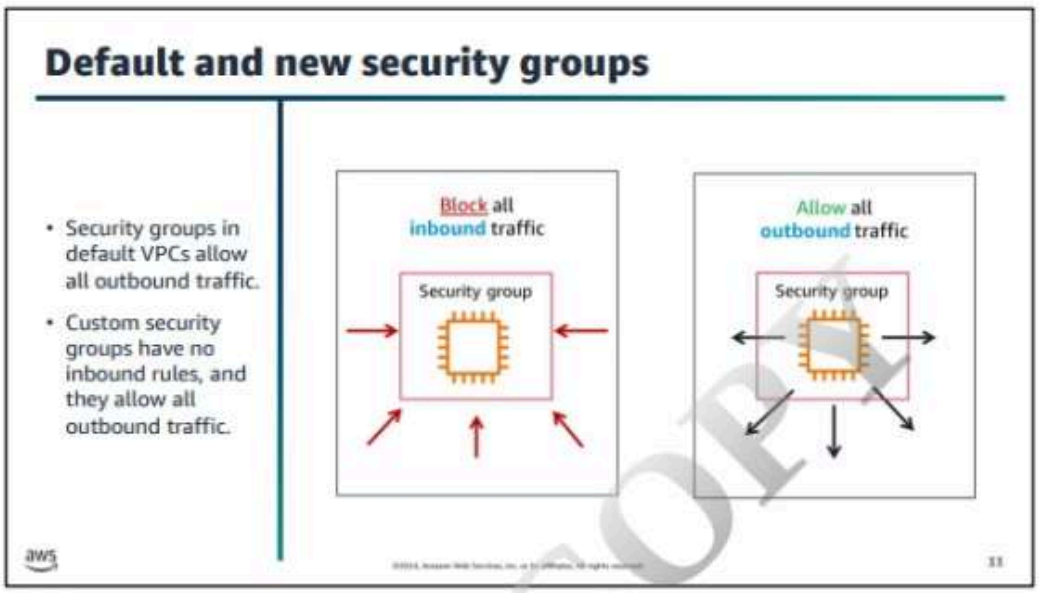
Consider the following example of stateful rules. Suppose that you initiate an Internet Control Message Protocol (ICMP) ping command to your instance from your home computer. If your inbound security group rules allow ICMP traffic, then information about the connection (including the port information) is tracked. Response traffic from the instance for the ping command is not tracked as a new request. It's tracked as an established connection, and is allowed to flow out of the instance, even if your outbound security group rules restrict outbound ICMP traffic.

Not all flows of traffic are tracked. A security group rule might permit TCP or UDP flows for all traffic (0.0.0.0/0). If it has a corresponding rule in the other direction that permits the response traffic, that flow of traffic is not tracked. The response traffic is therefore allowed to flow based on the inbound or outbound rule that permits the response traffic, and not on tracking information.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



A security group that is created with a default VPC includes an outbound rule that allows all outbound traffic. You can remove the rule and add outbound rules that allow specific outbound traffic only. If your security group has no outbound rules, no outbound traffic that originates from your instance is allowed. Traffic can be restricted by protocol, by service port, and by source IP address (individual IP address or CIDR block) or security group.

Security groups can be configured to set different rules for different classes of instances. Consider the case of a traditional three-tiered web application. The group for the web servers would have port 80 (HTTP) or port 443 (HTTPS) open to the internet. The group for the application servers would have port 8000 (application specific) accessible only to the web server group. The group for the database servers would have port 3306 (MySQL) open only to the application server group. All three groups would permit administrative access on port 22 (SSH), but only from the customer's corporate network. With this mechanism, you can deploy highly secure applications.

For more information about VPC security groups, see "Control traffic to your AWS resources using security groups" in the *Amazon Virtual Private Cloud User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/VPC_SecurityGroups.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Custom security group rules

Inbound

Source	Protocol	Port	Comments
0.0.0.0/0	TCP	80	Allows inbound HTTP access from all IPv4 addresses
0.0.0.0/0	TCP	443	Allows inbound HTTPS traffic from anywhere

Outbound

Destination	Protocol	Port	Comments
SG ID of DB servers	TCP	1433	Allows outbound Microsoft SQL Server access to instances in the specified security group
SG ID of MySQL servers	TCP	3306	Allows outbound MySQL access to instances in the specified security group

aws 34

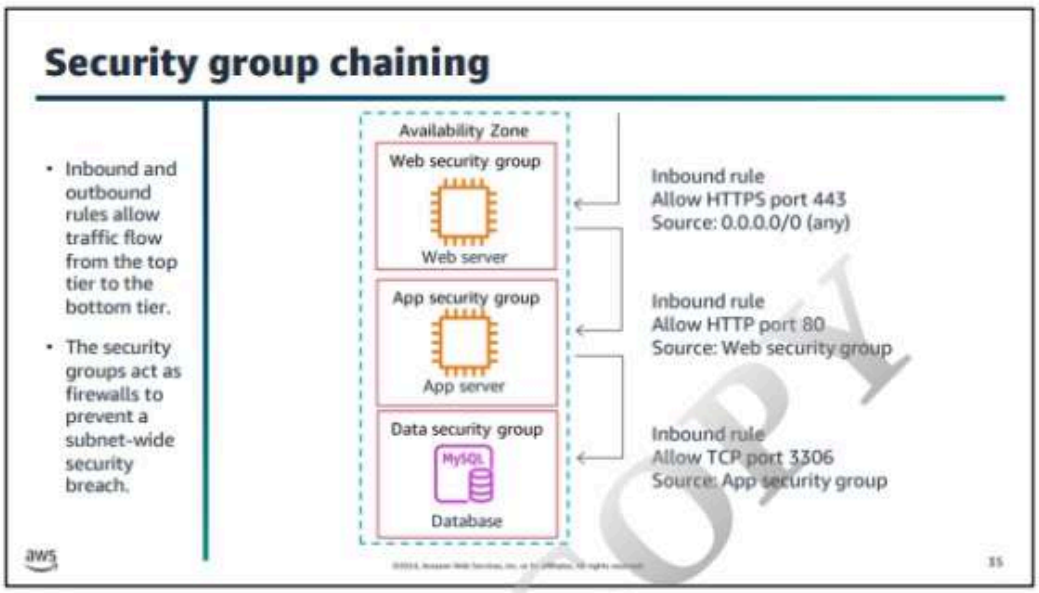
With security group rules, you can filter traffic based on protocols and port numbers. Security groups are stateful. If you send a request from your instance, the response traffic for that request is allowed to flow in regardless of inbound security group rules.

This table displays both inbound and outbound security group rules for a web server. The inbound rules allow for traffic on port 80 and port 443. Any user that requests the web server would be allowed in, and the web server would return the response back to their request. From the outbound perspective, if trying to send traffic, not in response to something that was requested on 480 or 443, you are limited to ports 1433 and 3306.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

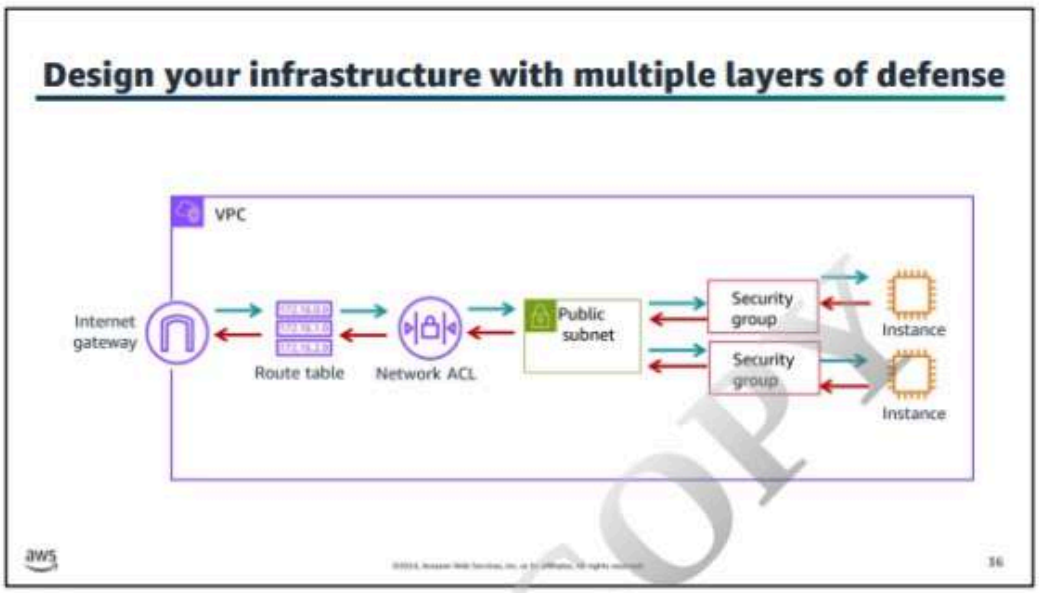


Here's an example of a chain of security groups. The inbound and outbound rules are set up so that traffic can only flow from the top tier to the bottom tier and back up again. The security groups act as firewalls to prevent a security breach in one tier from automatically providing subnet-wide access of all resources to the compromised client.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



As a best practice, you should secure your infrastructure with multiple layers of defense. You can control which instances are exposed to the internet by running your infrastructure in a VPC with a properly configured internet gateway and route tables. You can also define security groups and network ACLs to further protect your infrastructure at the interface and subnet levels. Additionally, you should secure your instances with a firewall at the operating system level and follow other security best practices.

AWS customers typically use security groups as their primary method of network packet filtering. Security groups are more versatile than network ACLs because of their ability to perform stateful packet filtering and to use rules that reference other security groups. However, network ACLs can be effective as a secondary control for denying a specific subset of traffic or providing high-level guardrails for a subnet.

By implementing both network ACLs and security groups as a defense in depth means of controlling traffic, a mistake in the configuration of one of these controls will not expose the host to unwanted traffic.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Comparing security groups and network ACLs

Security Group	Network ACL
Associated to an elastic network interface and implemented in the hypervisor	Associated to a subnet and implemented in the network
Supports Allow rules only	Supports Allow rules and Deny rules
A stateful firewall	A stateless firewall
All rules evaluated before deciding whether to allow traffic	All rules processed in order when deciding whether to allow traffic
Applies to an instance only if it is associated with the instance	Applies to all instances deployed in the associated subnet

17

A security group acts as a firewall for associated EC2 instances, controlling both inbound and outbound traffic at the instance level. Network ACLs act as a firewall for associated subnets, controlling both inbound and outbound traffic at the subnet level.

Both can have different default configurations depending on how they are created.

Security groups
- Security groups in default VPCs allow all traffic.
- New security groups have no inbound rules and allow outbound traffic.

Network ACLs
- Network ACLs in default VPCs allow all inbound and outbound IPv4 traffic.
- Custom network ACLs deny all inbound and outbound traffic, until you add rules.

DO NOT COPY javierartiga.espublico@gmail.com

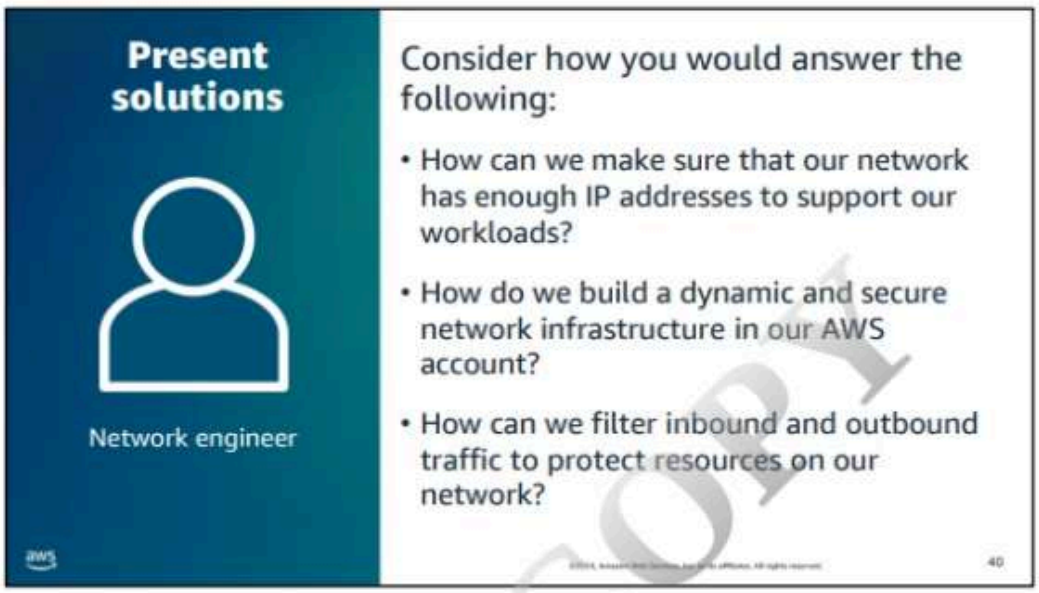
© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Imagine that you are now ready to talk to the network engineer and present solutions that meet their architectural needs. Think about how you would answer the questions from the beginning of the lesson.

Your answers should include the following solutions:

- With AWS, you define IP address ranges by using CIDR blocks. You can assign block sizes between /28 (16 IP addresses) and /16 (65,536 IP addresses) for IPv4 subnets.
- Build a dynamic and secure network infrastructure by using VPC components.
- Protect the network by filtering inbound and outbound traffic with network ACLs and security groups.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Module review

In this module, you learned about the following:

- ✓ IP addresses
- ✓ VPC fundamentals
- ✓ VPC traffic security

Next, you review the following:

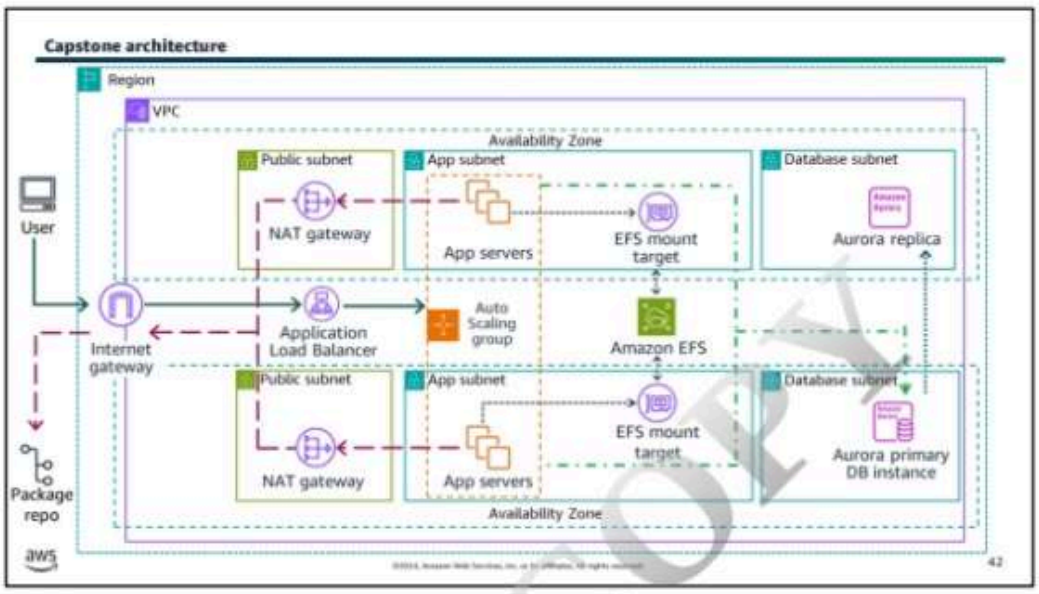
- Capstone check-in
- Knowledge check

41

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



You build this multi-tier architecture in the capstone lab.

Note: The diagram on this slide contains several abbreviations. The following list includes some services that are not covered in this module, but are included for completeness:

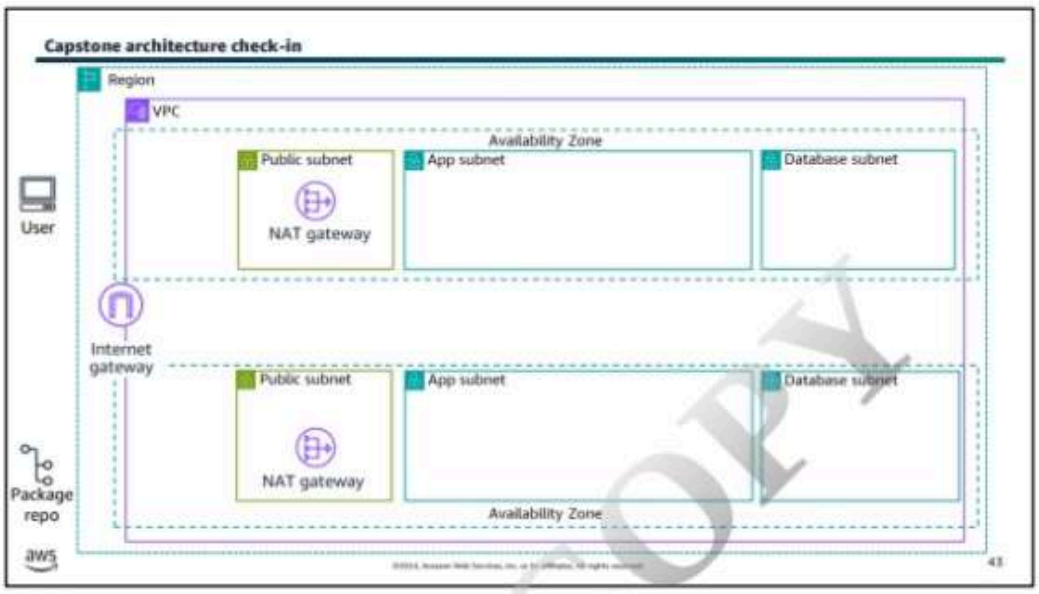
- EFS = Amazon Elastic File Service (Amazon EFS)
- Aurora = Amazon Aurora

[Image description: The diagram shows a single AWS Region with one VPC and two Availability Zones. Each Availability Zone contains a public subnet, an app subnet, and a database subnet. A unidirectional arrow outside the VPC routes from the remote user through an internet gateway, to an Application Load Balancer. The Application Load Balancer is depicted as not residing in a subnet or an Availability Zone, though it does reside in both public subnets. The arrow continues to the Auto Scaling group that has app servers in the app subnets of both Availability Zones. Each app server communicates with an EFS mount target in its own subnet to reach the EFS file system, which does not reside inside the VPC and is shown this way for simplicity. All app servers communicate with an Amazon Aurora primary DB instance in one of the database subnets. The other database subnet holds the Aurora replica. A unidirectional arrow points between the Aurora primary and Aurora replica. A unidirectional arrow points from the app servers and travels through each Availability Zone's NAT gateway. The NAT gateways are in the public subnet of each Availability Zone. The arrows travel through each NAT gateway, and exit the VPC through the internet gateway and reach the remote package repository. **Note:** The arrows in the diagram are shown as unidirectional to indicate where the request originated. Assume that the response from the target travels the path in reverse unless otherwise indicated. **End description.]**

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



In this module, you explored AWS networking services and resources.

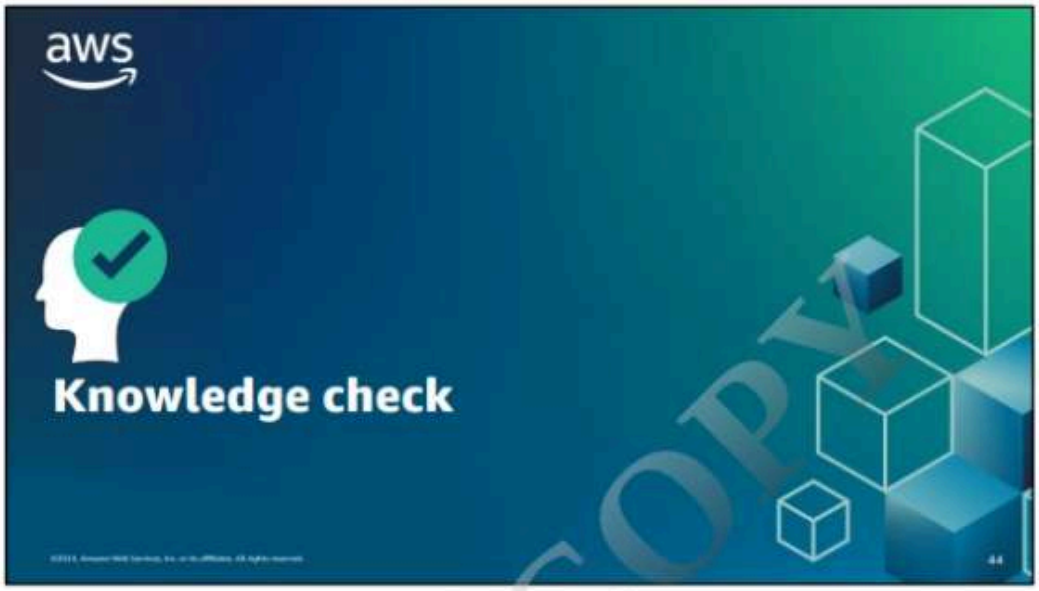
Review the capstone architecture to explore some of the design decisions. This architecture helps you provide the following benefits:

- You achieve high availability by setting up an Amazon VPC across two Availability Zones. If one Availability Zone stops working, you can direct traffic to the remaining Availability Zone.
- You protect resources by dividing them into separate subnets: public, application, and data. You can control the traffic that reaches each group of resources. You can also better isolate errors and vulnerabilities that occur in a subnet.
- You create one of each of these subnets in both Availability Zones.
- To allow internet access on your network, you set up an internet gateway. The internet gateway also protects private IP addresses of resources on your network. The internet gateway performs network address translation between public and private IP addresses.
- You set up NAT gateways in the public subnets to handle outbound traffic to the internet from private subnets. This provides internet connectivity while preventing external traffic from connecting with your private instances.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 1

True or False: A single Amazon Virtual Private Cloud (Amazon VPC) can span multiple Regions.

Choice	Response
A	True
B	False

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 1 and answer

True or False: A single Amazon Virtual Private Cloud (Amazon VPC) can span multiple Regions.

Choice	Response
A	True
B correct	False

The correct answer is B, false.

An Amazon VPC spans all of the Availability Zones in one Region.

For more information, see "Configure a virtual private cloud" in the *Amazon Virtual Cloud User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/configure-your-vpc.html.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 2

Which action must be taken to make a subnet public?

Choice	Response
A	Route outbound traffic from the subnet.
B	Route inbound traffic from the internet gateway.
C	Route outbound traffic to the internet gateway.
D	Subnets are public by default.

47

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 2 and answer

Which action must be taken to make a subnet public?

Choice	Response
A	Route outbound traffic from the subnet.
B	Route inbound traffic from the internet gateway.
C correct	Route outbound traffic to the internet gateway.
D	Subnets are public by default.

The correct answer is C, route outbound traffic to the internet gateway.

In your public subnet's route table, you can specify a route for the internet gateway to all destinations not explicitly known to the route table (0.0.0.0/0 for IPv4 or ::/0 for IPv6). Alternatively, you can scope the route to a narrower range of IP addresses, for example, the public IPv4 addresses of your company's public endpoints outside of AWS, or the Elastic IP addresses of other Amazon EC2 instances outside of your VPC.

For more information, see "Enable VPC internet access using internet gateways" in the *Amazon VPC User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/VPC_Internet_Gateway.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 3

Which function does the NAT gateway serve?

Choice	Response
A	Load balances incoming traffic to multiple instances
B	Allows internet traffic initiated by private subnet instances
C	Allows instances to communicate between subnets
D	Increases security for instances in a public subnet

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 3 and answer

Which function does the NAT gateway serve?

Choice	Response
A	Load balances incoming traffic to multiple instances
B correct	Allows internet traffic initiated by private subnet instances
C	Allows instances to communicate between subnets
D	Increases security for instances in a public subnet

50

The correct answer is B, allows internet traffic initiated by private subnet instances.

You can use a NAT device to allow instances in private subnets to connect to the internet, other VPCs, or on-premises networks. These instances can communicate with services outside the VPC, but they cannot receive unsolicited connection requests.

For more information about NAT gateways, see "Connect to the internet or other networks using NAT devices" in the *Amazon VPC User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/vpc-nat.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 4

What should be used to create traffic filtering rules for a subnet?

Choice	Response
A	NAT gateway
B	Route table
C	Security group
D	Network access control list (network ACL)

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 4 and answer

What should be used to create traffic filtering rules for a subnet?

Choice	Response
A	NAT gateway
B	Route table
C	Security group
D correct	Network access control list (network ACL)

The correct answer is D, network ACL.

A network ACL contains a numbered list of rules. You evaluate the rules in order, starting with the lowest numbered rule. The purpose is to determine whether traffic is allowed in or out of any subnet that is associated with the network ACL.

For more information about network ACLs, see "Control subnet traffic with network access control lists" in the Amazon VPC User Guide at https://docs.aws.amazon.com/vpc/latest/userguide/vpc-network-acls.html.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 5

Which ports are open by default when a new security group is created? (Select TWO.)

Choice	Response
A	Nothing is allowed inbound.
B	Nothing is allowed outbound.
C	Anything is allowed inbound.
D	Anything is allowed outbound.
E	Inbound traffic is allowed on public subnets.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Knowledge check question 5 and answer

Which ports are open by default when a new security group is created? (Select TWO.)

Choice	Response
A correct	Nothing is allowed inbound.
B	Nothing is allowed outbound.
C	Anything is allowed inbound.
D correct	Anything is allowed outbound.
E	Inbound traffic is allowed on public subnets.

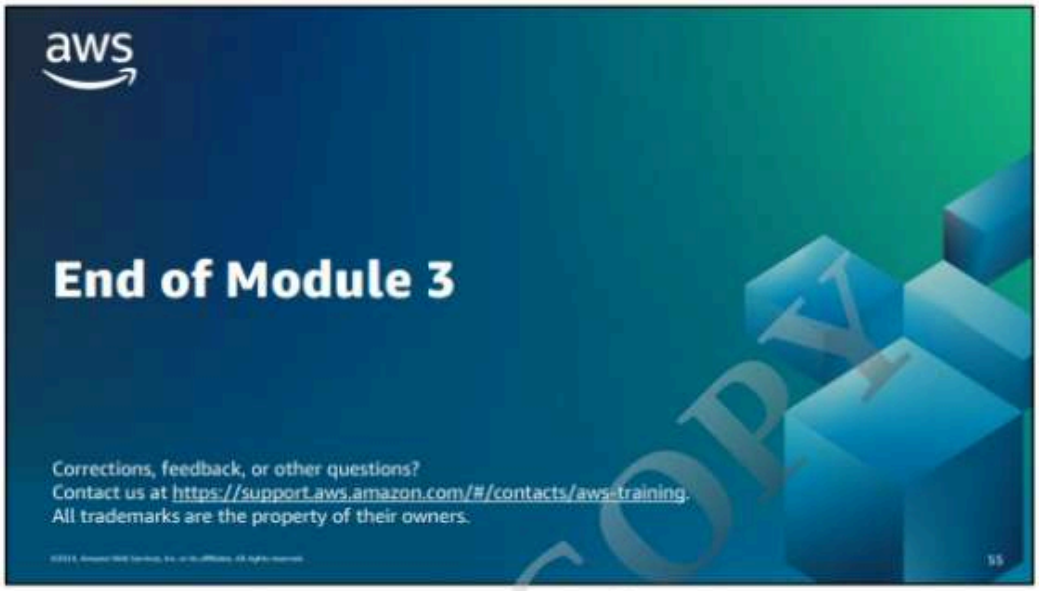
The correct answers are A, nothing allowed inbound, and D, anything allowed outbound.

Nothing is allowed inbound, and anything is allowed outbound. New security groups have no inbound rules and allow outbound traffic.

For more information about security groups, see "Control traffic to your AWS resources using security groups" in the *Amazon VPC User Guide* at https://docs.aws.amazon.com/vpc/latest/userguide/VPC_SecurityGroups.html.

DO NOT COPY javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

**Accessibility note for screen reader users: To make code examples in this presentation more accessible, Start of code and End of code labels have been added before and after blocks of code on slides in this presentation.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Poll question

Where do you run the majority of your compute workloads?

A. On-premises physical or virtual servers

B. Cloud-based servers

C. On-premises containers

D. Cloud-based containers

DO NOT COPY
javierartiga.espublico@gmail.com

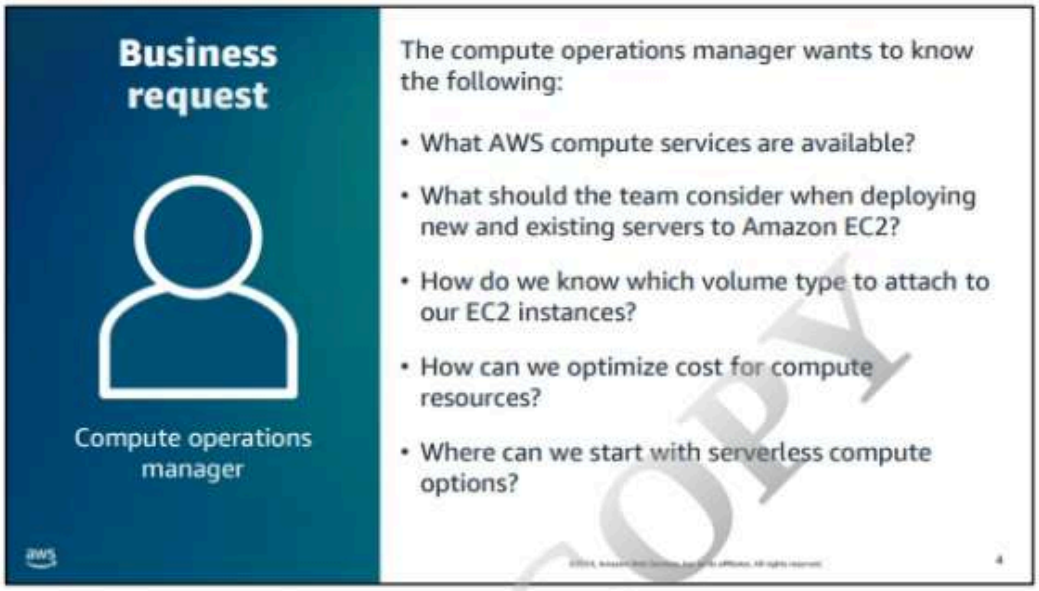
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Module overview

- Business request
- Compute services
- Amazon Elastic Compute Cloud (Amazon EC2) instances
- EC2 instance storage
- Amazon EC2 pricing options
- AWS Lambda
- Present solutions
- Knowledge check
- Capstone check-in
- Lab 2: Build Your Amazon VPC Infrastructure

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Imagine that your compute operations manager contacted you with questions about moving and building their workloads on Amazon Web Services (AWS). Here are some questions that they are asking.

At the end of this module, you meet with the compute operations manager and present some solutions.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

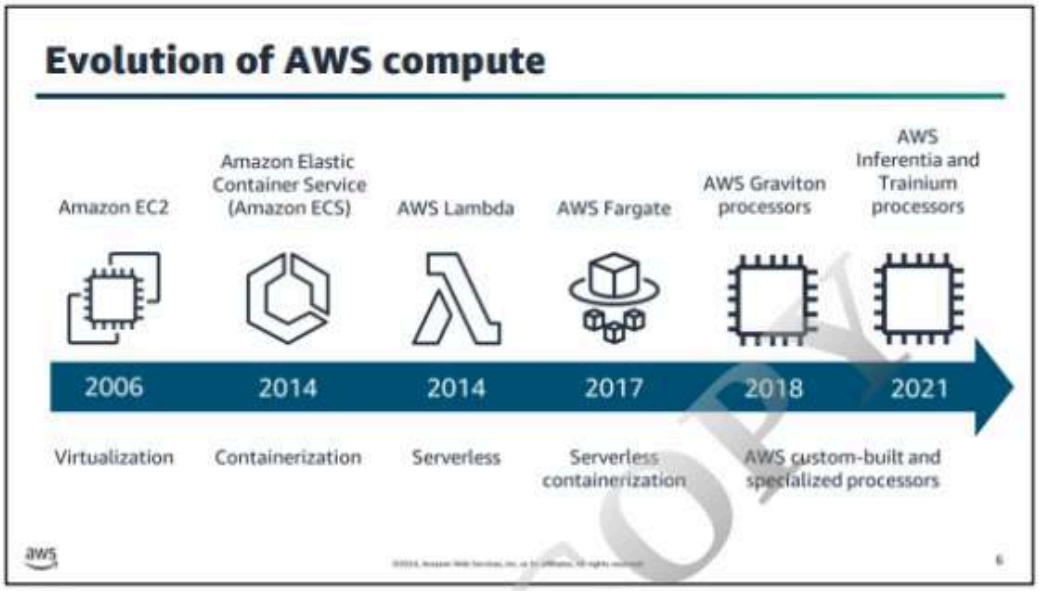
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

The compute operations manager asks, "What AWS compute services are available?" They need your help to find the best service to fit their use case.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Virtual machines (VMs) provide the following benefits:

- Hardware independence
- Faster provisioning speed, in minutes or hours
- Pay-as-you-go pricing models instead of hardware purchases
- More scale
- Elastic resources
- Greater agility
- Reduced maintenance

Amazon EC2 was one of the first AWS services released in 2006, and it continues to be a central component of cloud computing. New generations of EC2 instance types introduce greater compute efficiency that can help reduce compute costs.

Containerization provides the following benefits:

- Platform independence
- A consistent runtime environment
- Higher resource use
- Easier and faster deployments
- Isolation and sandboxing
- Quicker start speed, so you can deploy in seconds

In 2014, Amazon Elastic Container Service (Amazon ECS) introduced the ability to run distributed applications on a managed cluster EC2 instance with Docker containers. Support for Kubernetes was released in 2017 with Amazon Elastic Kubernetes Service (Amazon EKS).

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Serverless computing provides the following benefits:

- Continuous scaling
- Built-in fault tolerance
- Pay for value
- Zero maintenance

AWS Lambda also appeared in 2014, introducing serverless computing. With Lambda, you can run code without provisioning or managing EC2 instances. Serverless computing and containerization were combined in 2017 with the release of AWS Fargate. AWS Fargate is a serverless compute engine for containers that works with Amazon ECS and Amazon EKS.

AWS has also custom-built processors and introduced a variety of AWS Graviton processors. Graviton processors are built around Arm cores and make extensive use of silicon. In 2018, AWS Graviton was released and built for scale-out workloads where you can share the load across a group of smaller instances. In 2020, AWS Graviton2 was released. It uses 64-bit Arm Neoverse cores to provide the best price performance for cloud workloads that run in Amazon EC2. In 2022, AWS Graviton3 was released to provide the best price performance for workloads in Amazon EC2. In 2024, AWS Graviton4 was released to deliver the best performance and energy efficiency for a broad range of workloads running on Amazon EC2.

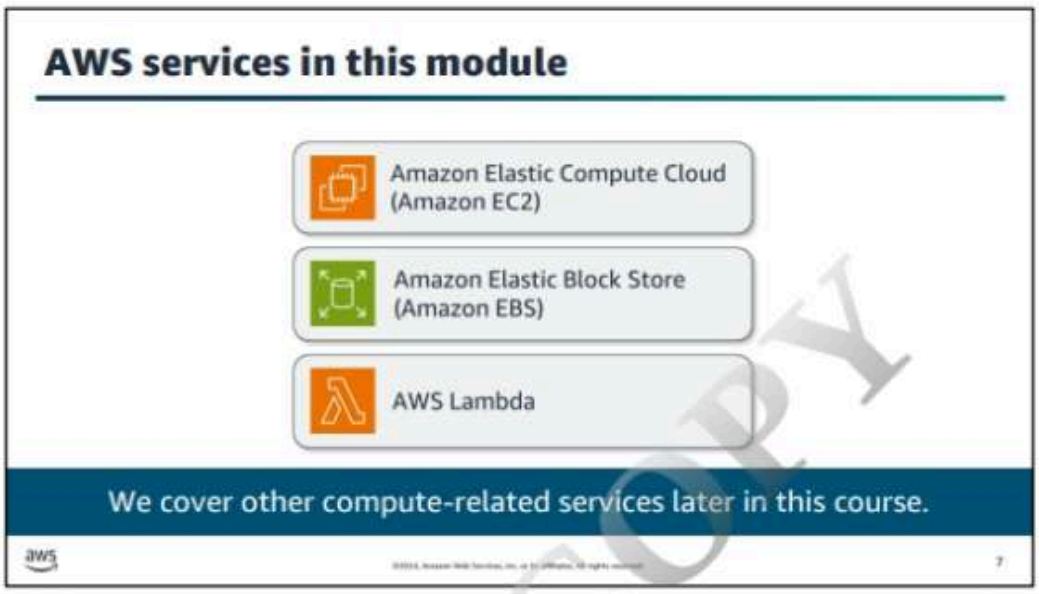
AWS introduced specialized processors to support the implementation of AI and machine learning (ML). AWS Inferentia chips were introduced in 2019 to provide high-performance machine learning inference chips, which AWS designed and built. These chips were created to support ML inference applications. In 2023, AWS Inferentia2 was introduced, delivering up to 40% better price performance than other comparable Amazon EC2 instances.

AWS Trainium was introduced in 2021 as the second ML chip that AWS built. Trainium is optimized for high-performance deep learning training.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



In this module, you learn about Amazon EC2, Amazon EBS, and AWS Lambda.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

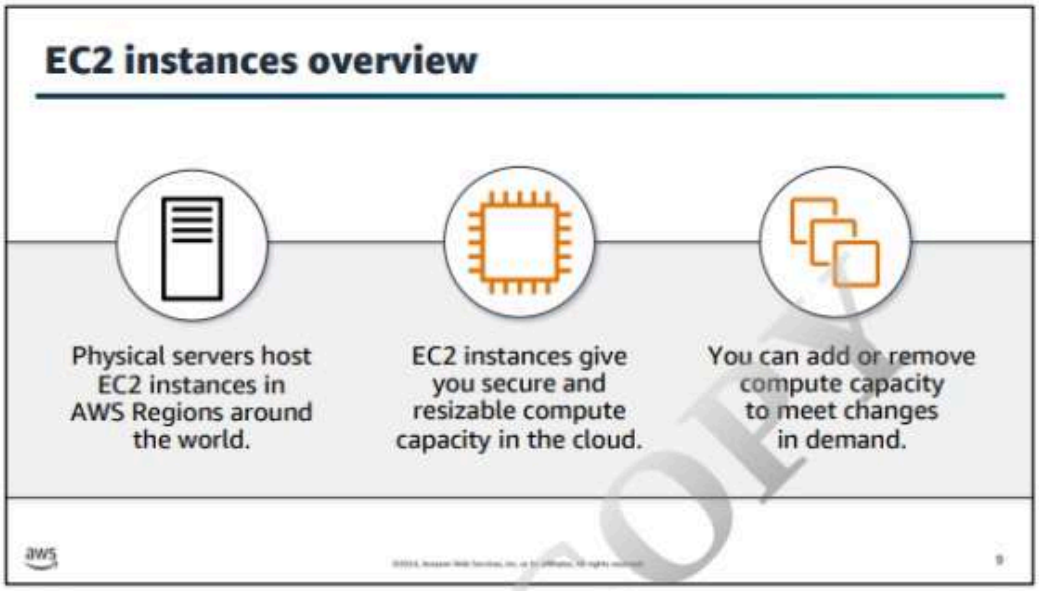
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

The compute operations manager asks, "What should the team consider when deploying new and existing servers to Amazon EC2?" The team needs information about EC2 instances. They need you to tell them what is required at launch and which advanced settings they should explore.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



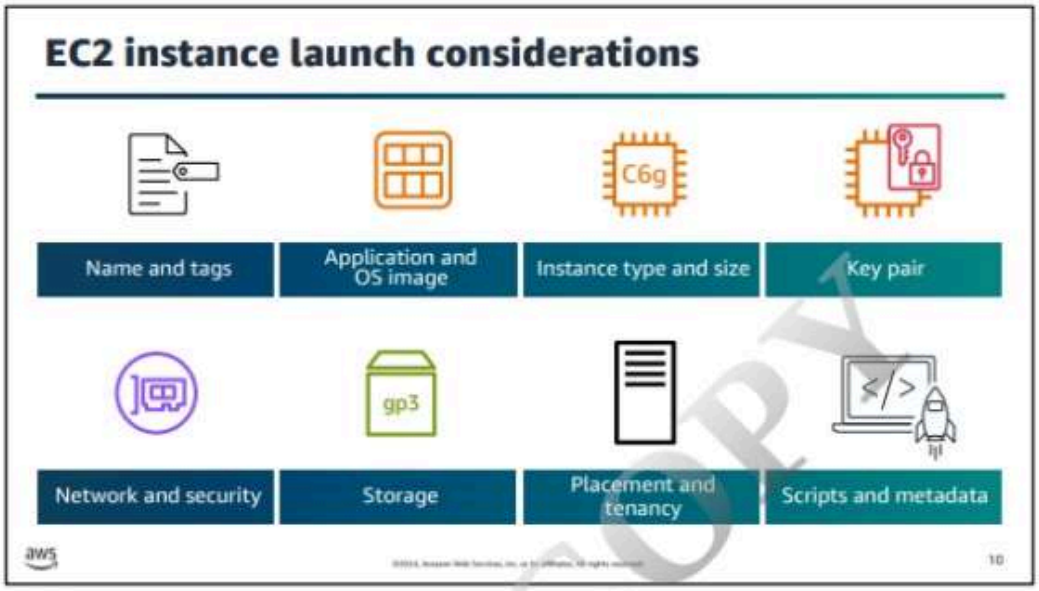
Amazon EC2 is the service that you use to create and run virtual machines. Your virtual machines in the AWS Cloud are called EC2 instances. Amazon EC2 is just like your traditional on-premises server, but it is available in the cloud. It can support workloads such as web hosting, applications, databases, authentication services, and anything else that a server can support.

On AWS, servers, databases, storage, and higher-level application components can be instantiated within seconds. You can treat these resources as temporary and disposable, free from the constraints of a fixed IT infrastructure. Elastic cloud computing redefines the way that you approach change management, testing, reliability, and capacity planning.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



You must consider many options before you start working with EC2 instances.

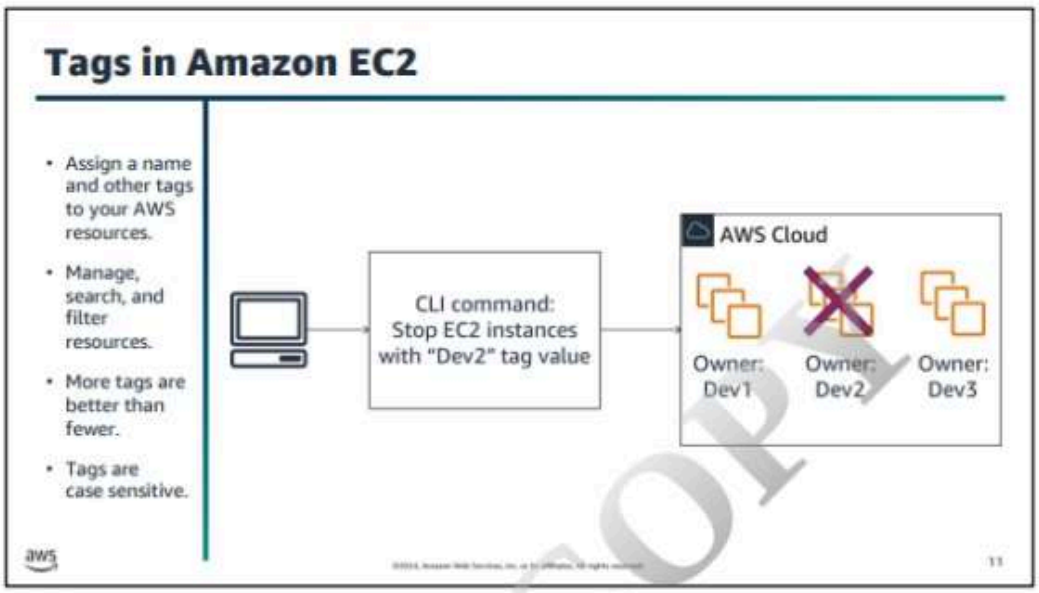
Consider the following options:

- **Name and tags** – How should your instance be identified?
- **Application and operating system (OS) image** – What will you start running?
- **Instance type and size** – What technical requirements do you have?
- **Authentication and key pair** – How do you plan to connect to the instance?
- **Network settings and security** – What virtual private cloud (VPC), subnet, and security groups will you use?
- **Configure storage** – What type of block storage is best for your use case?
- **Placement and tenancy** – Where should you run your EC2 instances?
- **Scripts and metadata** – What can you do to automate your launch?

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



On AWS, you can assign metadata to your resources in the form of *tags*. Each tag is a simple label that consists of a customer-defined key and an optional value. Use tags to filter resources. Tags can manage resource access control, track costs, help automate tasks, and keep you organized. Although no tag types are required, you can create tags to categorize resources by purpose, owner, environment, or other criteria.

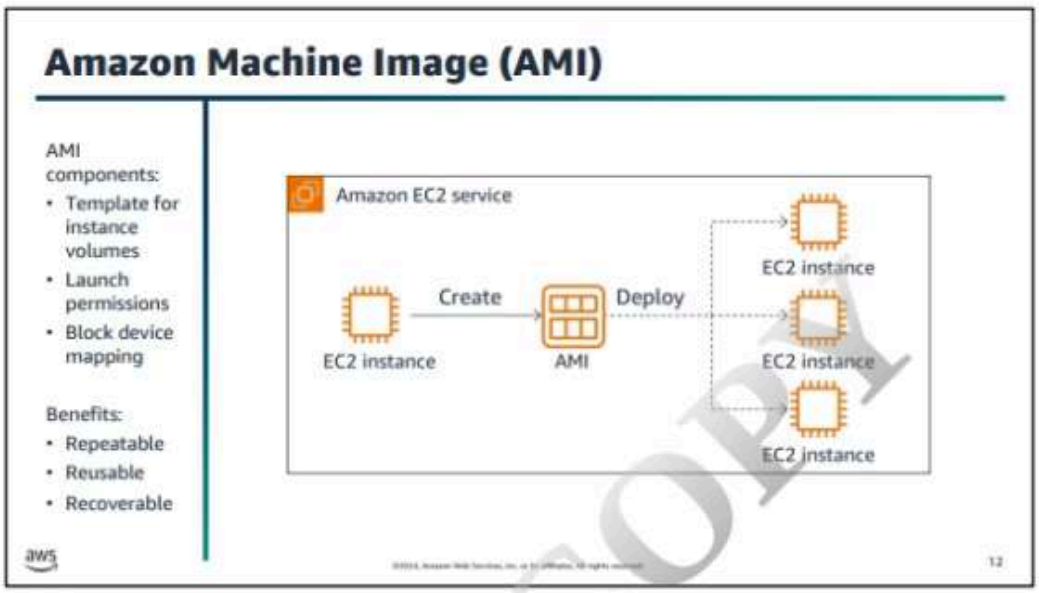
In the example, each instance has a tag with key "Owner" and values of "Dev1," "Dev2," or "Dev3." A user sends n AWS Command Line Interface (AWS CLI) command from a terminal. This CLI command will stop all EC2 instances in a Region that have a key-value pair of Owner:Dev2. Remember that your tags are case sensitive. Your workflows that involve tag key-value pairs need an exact string match to work properly.

For more information about AWS tagging strategies, see "What is Tag Editor?" in the *Tagging AWS Resources and Tag Editor User Guide* at https://docs.aws.amazon.com/general/latest/gr/aws_tagging.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



An Amazon Machine Image (AMI) provides the information that is required to launch an instance, which is a virtual server in the cloud. When you need multiple instances with the same configuration, you can launch multiple instances from a single AMI. When you need instances with different configurations, you can use different AMIs to launch instances. Specify a source AMI when you launch an instance.

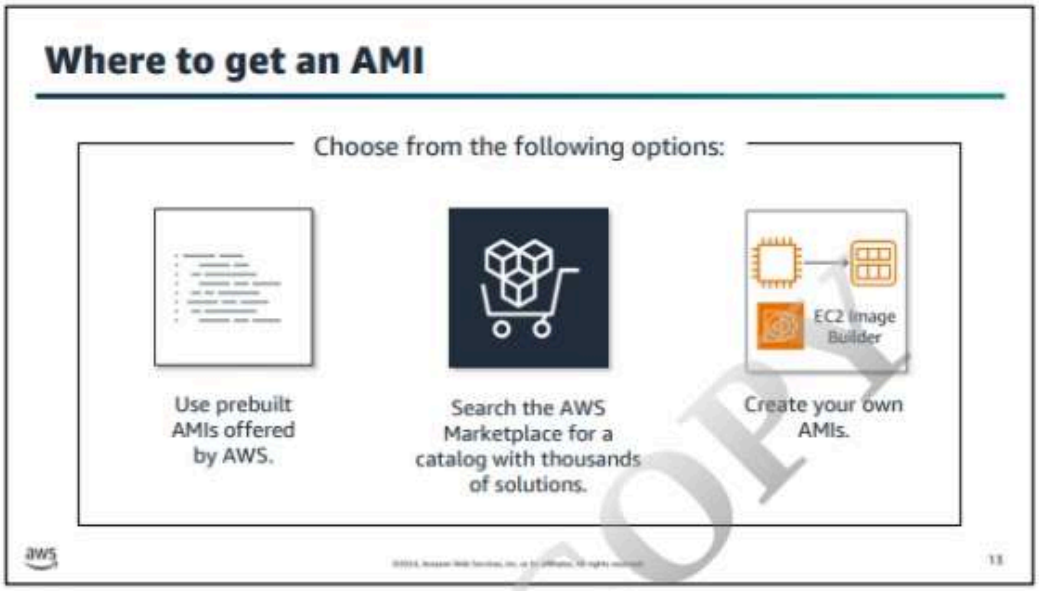
An AMI includes the following attributes:

- A template for the root volume for the instance (for example, an OS, an application server, and applications)
- Launch permissions that control which AWS accounts can use the AMI to launch instances
- Block device mapping that specifies the volumes to attach to the instance when it's launched

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



You can use AMIs that AWS provides, or you can create your own custom AMIs.

You can buy or sell AMIs through the following sources:
- AWS user community
- AWS Marketplace

Alternatively, you can choose one of your own custom AMIs and share it with other AWS accounts. Some organizations create custom AMIs to speed up deployment based on internal requirements. Tools that are used in customization include Chef, Puppet, and cloud-init.

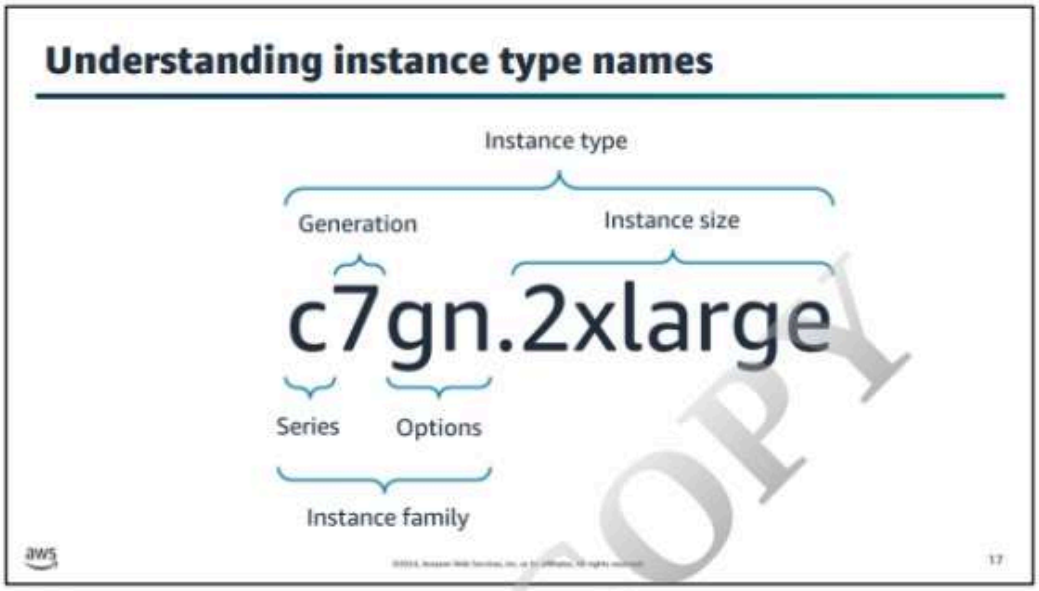
To create a custom AMI, launch an instance and customize it to meet your requirements. Then, save that configuration as a custom AMI. Instances that are launched from this custom AMI will use all of your customizations. These custom AMIs can also be published for internal, private, or external public use. As a publisher of an AMI, you are responsible for the initial security posture of the machine images that you use in production.

For more information about AMIs, see "Amazon Machine Images in Amazon EC2" in the *Amazon Elastic Compute Cloud User Guide* at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AMIs.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



You can choose from more than 800 types of EC2 instances to run applications that you move to the cloud. Each instance type comes in different sizes, with different allotments of virtual CPUs (vCPUs) and memory. Choosing the right instance size is critical to using them efficiently. An instance's full type consists of its family name, followed by the generation number, any additional properties, and then the size.

In the example, explore the following pieces of the instance type name:

- **c** – The first position of the instance family indicates the *series*.
- **7** –The next number is the generation, which gradually increases as AWS upgrades hardware in its data centers.
- **g** – Sometimes one or more letters appear after the generation. The letters represent additional properties. In this example, the g stands for Graviton, an ARM-based processor that AWS developed.
- **n** – This represents another option, network and Amazon EBS-optimized.
- **2xlarge** – The last part represents the size of the instance, which includes the CPU, memory, storage, and network performance.

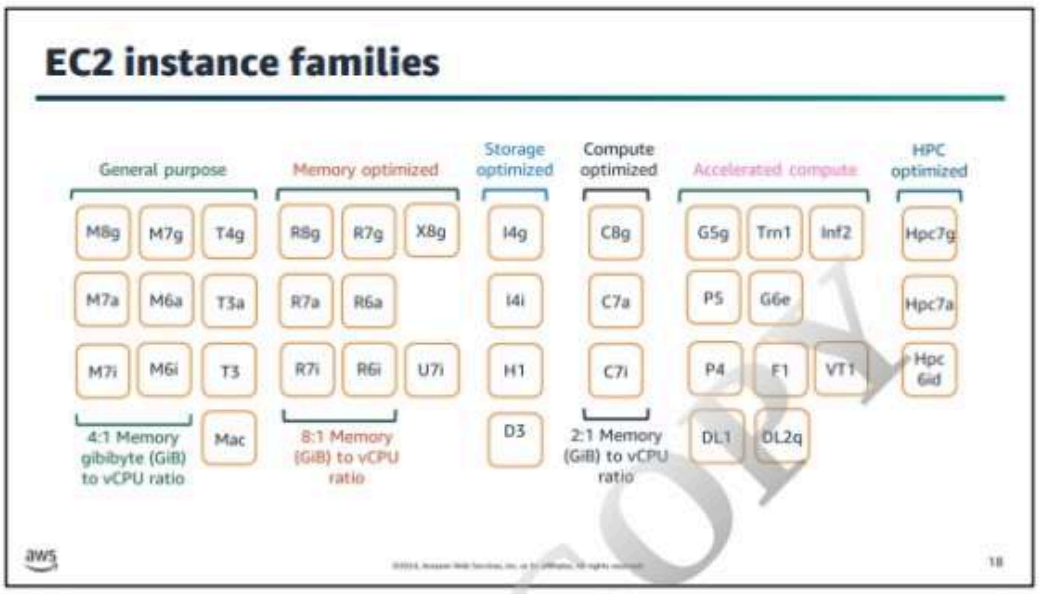
This instance type is read as a c-type instance from the seventh generation, with a Graviton3E processor and a 2 extra-large size deployment.

For more information about the series and options for instance types, see "Amazon EC2 instance type naming conventions" at https://docs.aws.amazon.com/ec2/latest/instancetypes/instance-type-names.html.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Pick the optimal instance family for the type of workload that you plan to deploy. This step saves time and cost, and reduces the need to resize later. Some instance types are only available in certain Regions.

General purpose
- Balance of compute, memory, and networking
- Diverse workloads
- Web applications

Memory optimized
- Fast delivery of large data sets in memory
- Database servers
- Web caches
- Data analytics

Storage optimized
- High sequential read/write
- Large data sets
- NoSQL databases
- Amazon OpenSearch Service

Compute optimized
- Compute-bound applications
- High-performance processors
- Media transcoding
- Scientific modeling
- Machine learning

Accelerated computing
- High-graphics processing

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

- Graphics processing unit (GPU) bound
- Machine learning
- High performance computing (HPC)
- Autonomous vehicles

HPC Optimized

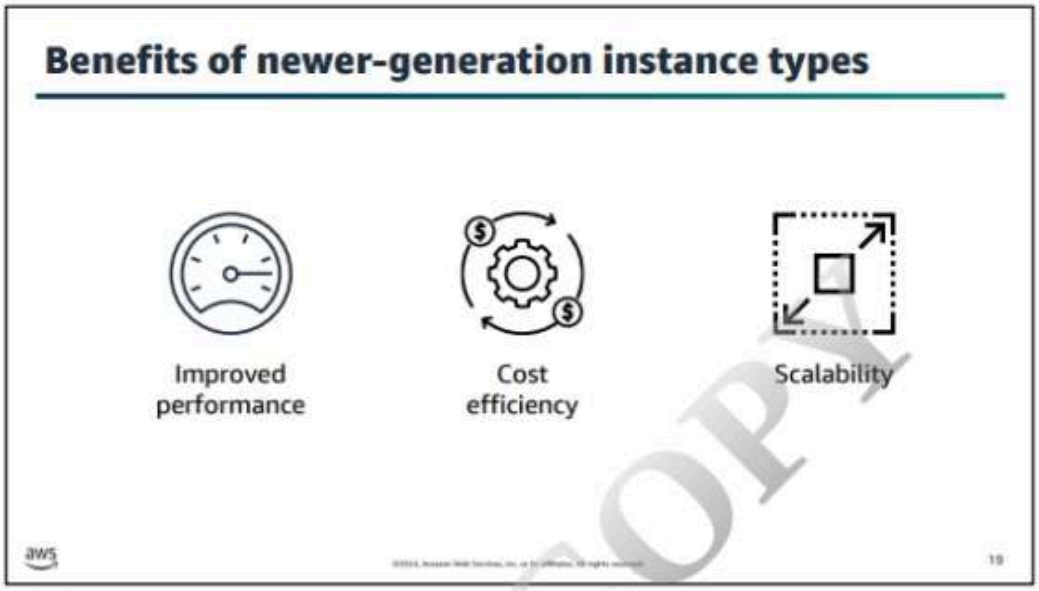
- Large, complex simulations
- Deep learning workloads

The instance types that are listed here are only a subset of what you can choose from in Amazon EC2. For more information about instance types, see Amazon EC2 Instance types at https://aws.amazon.com/ec2/instance-types/.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Next-generation EC2 instances generally increase compute capabilities and reduce processing costs that are associated with running the instance. Use the latest generation of instances to get the best performance while saving on compute costs.

Improved performance

- Newer hardware: Newer instances are built on the latest hardware, offering better CPU performance, faster memory, and improved storage options.
- Optimized networking: Newer instances often come with enhanced network capabilities, such as higher bandwidth and lower latency, improving overall application performance.

Cost efficiency

- Price-to-performance ratio: Newer instances typically provide more compute power or memory for the same or lower price compared to older generations, allowing you to get more value for your money.
- Energy efficiency: Newer instances often consume less power, resulting in lower operating costs for AWS, which can translate to cost savings for customers.

Scalability

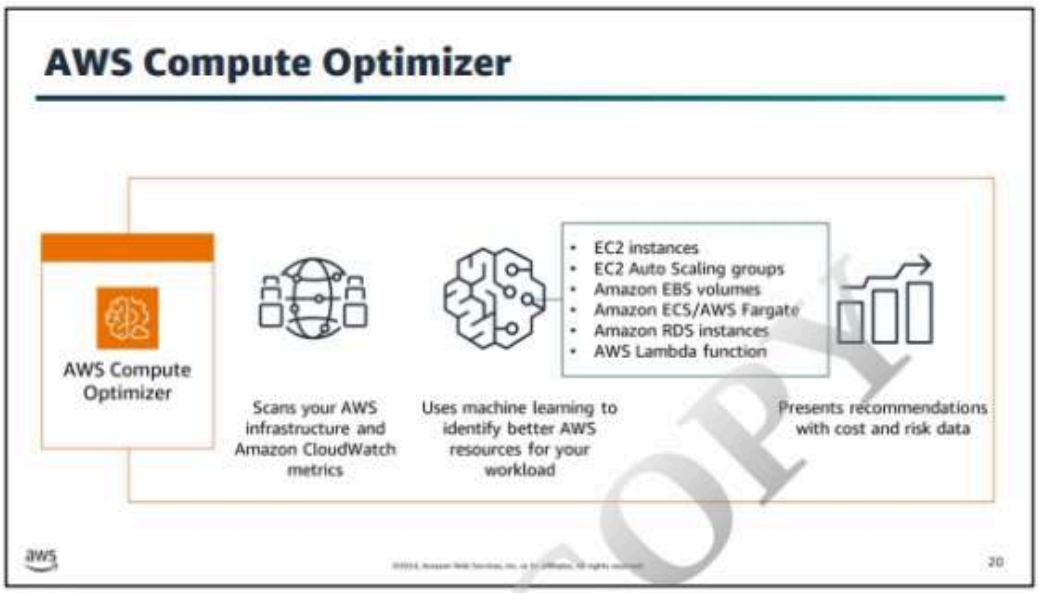
- More options: New instance families often offer more variety in terms of CPU, memory, and storage configurations, making it easier to select the right instance type for your workload.
- Future-proofing: Newer instances allow you to be better prepared for scaling and adapting to future AWS services and enhancements.

To explore the different instance types, see Instance Type Explorer at https://aws.amazon.com/ec2/instance-explorer/.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



AWS Compute Optimizer recommends more efficient AWS Compute resources for your workloads to reduce costs and improve performance. It does this by using machine learning to analyze historical utilization metrics.

Compute Optimizer generates recommendations for the following resources:

- Amazon EC2 instances
- Amazon EC2 Auto Scaling groups
- Amazon Elastic Block Store (Amazon EBS) volumes
- AWS Lambda functions
- Amazon Elastic Container Service (ECS) services on AWS Fargate
- Amazon Relational Database Service (RDS) DB instance classes

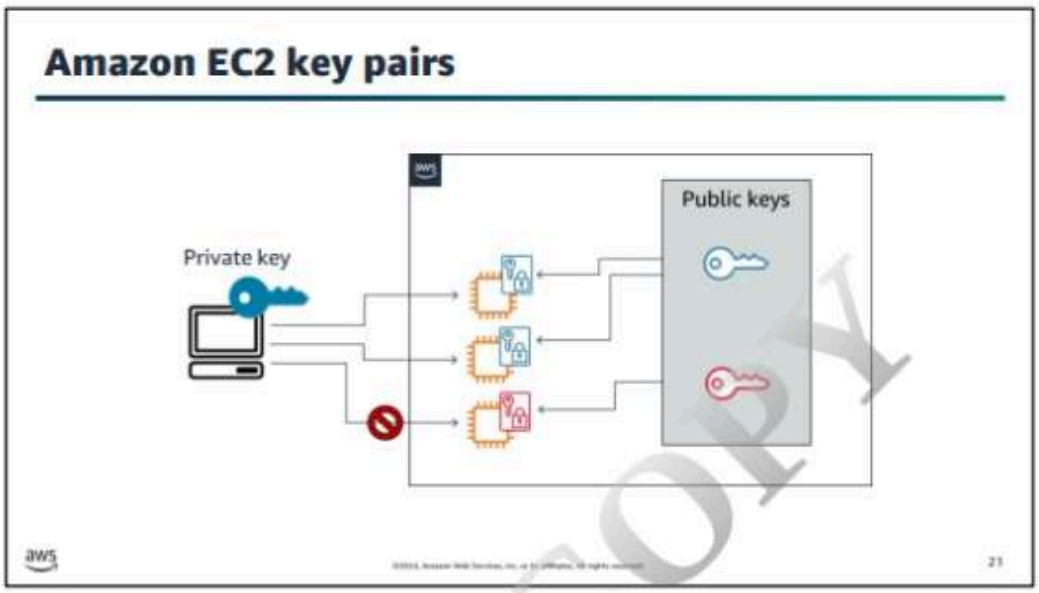
Over-provisioning compute can lead to unnecessary infrastructure cost. Under-provisioning compute can lead to poor application performance.

For more information, see "What is AWS Compute Optimizer?" in the *AWS Compute Optimizer User Guide* at https://docs.aws.amazon.com/compute-optimizer/latest/ug/what-is-compute-optimizer.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

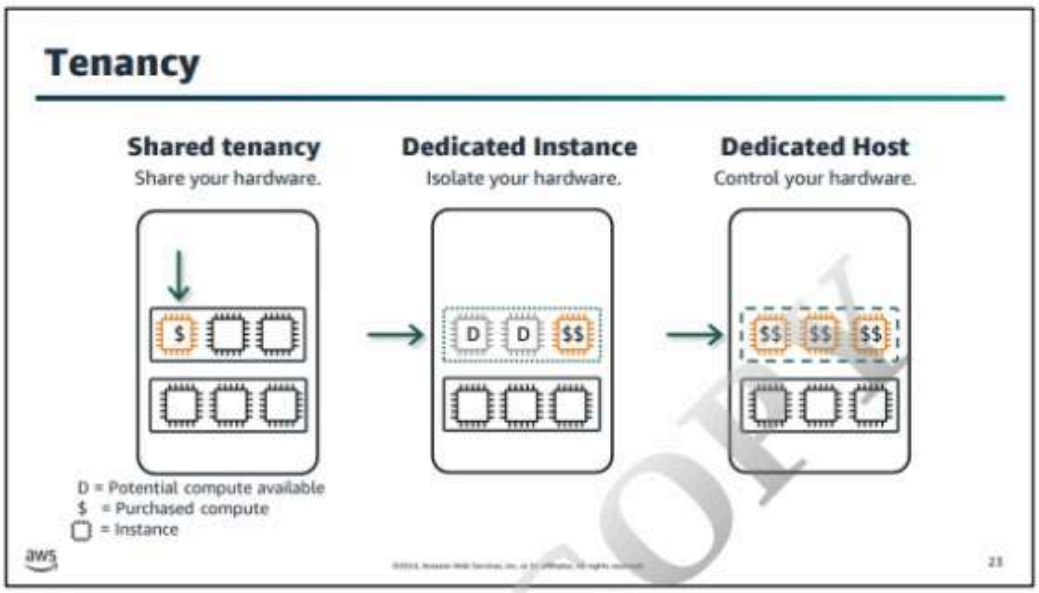
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



A *key pair*, which consists of a private key and a public key, is a set of security credentials. You use a key pair to prove your identity when connecting to an instance. Amazon EC2 stores the public key and you store the private key. You use the private key instead of a password to securely access your instances. Anyone who possesses your private keys can connect to your instances, so it's important that you store your private keys in a secure place.

DO NOT COPY
javierartiga.espublico@gmail.com

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



By default, EC2 instances have *shared tenancy*, which means that multiple AWS accounts might share the same physical hardware.

Dedicated Instances are EC2 instances that are physically isolated at the host hardware level. They are isolated from instances that aren't dedicated and from instances that belong to other AWS accounts. When you launch instances on a *Dedicated Host*, the instances run on a physical server with EC2 instance capacity fully dedicated to your use. You are provided an isolated server with configurations that you can control. With Dedicated Hosts, you have the option to have AWS automatically select a server to place your instance. Or, you can manually select a dedicated server to place your instance. Thus, you can deploy instances by using configurations to address corporate compliance and regulatory requirements.

With Dedicated Hosts, you can use your existing per-socket, per-core, or per-VM software licenses. These software licenses are bound to VMs, sockets, or physical cores, subject to your license terms, and include, among others, the following:

- Microsoft Windows Server
- Microsoft SQL Server
- SUSE Linux Enterprise Server
- Red Hat Enterprise Linux

For more information about Dedicated Hosts, see Amazon EC2 Dedicated Hosts at https://aws.amazon.com/ec2/dedicated-hosts/.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Placement groups and use cases

With **placement groups**, you choose how close or far your instances are from each other.

Requirements	Solution	Example use case
Provide low network latency and high network throughput.	**Cluster** EC2 instances near each other.	High performance computing (HPC)
Critical instances must be fault-tolerant.	**Spread** across network segments and racks.	Medical health record system
Avoid correlated hardware failures.	**Partition** in logical groups on separate hardware.	Large distributed and replicated workloads such as Apache Kafka, Apache Hadoop, and Apache Cassandra

24

The Amazon EC2 service attempts to spread out all of your instances across underlying hardware to minimize correlated failures. You can use *placement groups* to influence the placement of a set of interdependent instances to meet the needs of your workload.

Cluster placement groups are recommended for your applications that benefit from low network latency, high network throughput, or both. These groups are also recommended when the majority of the network traffic is between the instances in your group. HPC workloads can require this level of connectivity in your VPC.

Spread placement groups are recommended for applications that have a small number of critical instances that should be kept separate from each other. Services that require maximum uptime, such as a medical health record system, are more fault-tolerant in a spread.

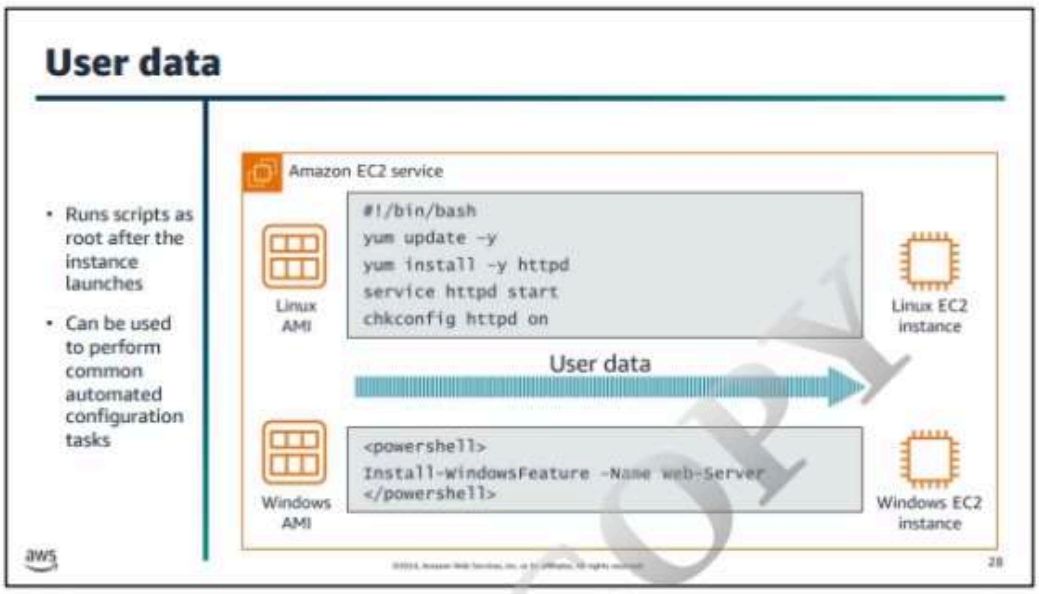
Partition placement groups can be used to deploy your large distributed and replicated workloads. You can use partitions to avoid same-time hardware failures for multiple components.

You can find examples of different types of placement groups in the documentation. For more information, see "Placement groups for your Amazon EC2 instances" in the *Amazon Elastic Compute Cloud User Guide* at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/placement-groups.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Amazon EC2 service running Linux and Windows instances. User data installs HTTP server on Linux and Web Server role on Windows during instance launch.
When creating your EC2 instances, you have the option of passing user data to the instance. User data can automate the completion of the instance launch. For example, user data might patch and update the instance AMI, fetch and install software license keys, or install additional software. User data is implemented as a shell script or cloud-init directive that runs with root or administrator privileges. It runs after the instance launches but before it becomes accessible on the network.

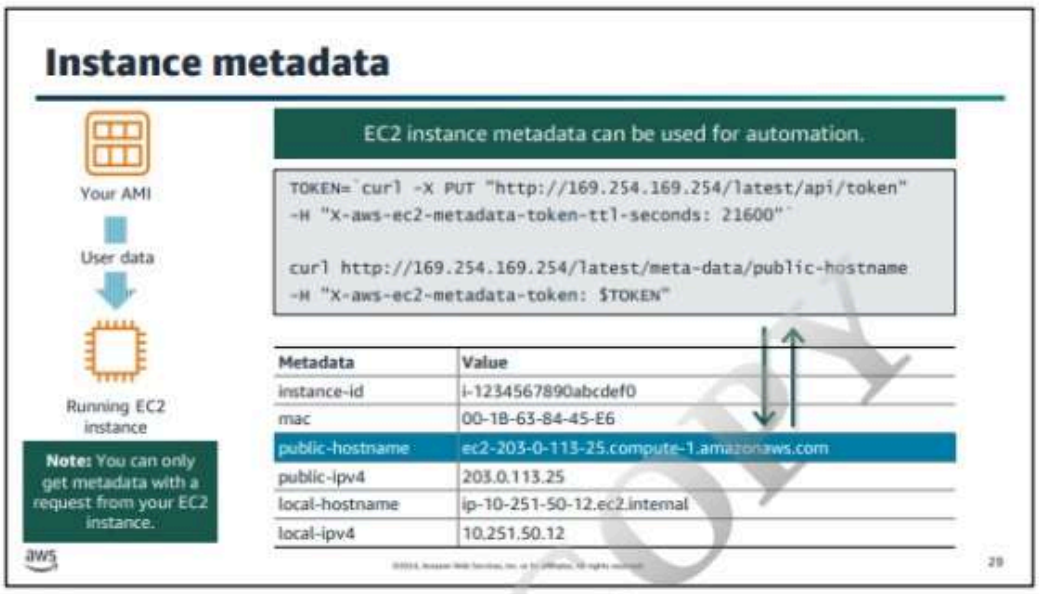
- User data is run by **cloud-init** on Linux.
- User data is run by **EC2Launch** service on Windows.

You can specify both a batch script and a Windows PowerShell script. Then, the batch script runs first and the Windows PowerShell script runs next, regardless of the order that they appear in the instance user data.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Instance metadata is data about your instance that you can use to configure or manage the running instance. Instance metadata is divided into categories, for example, host name, events, and security groups.

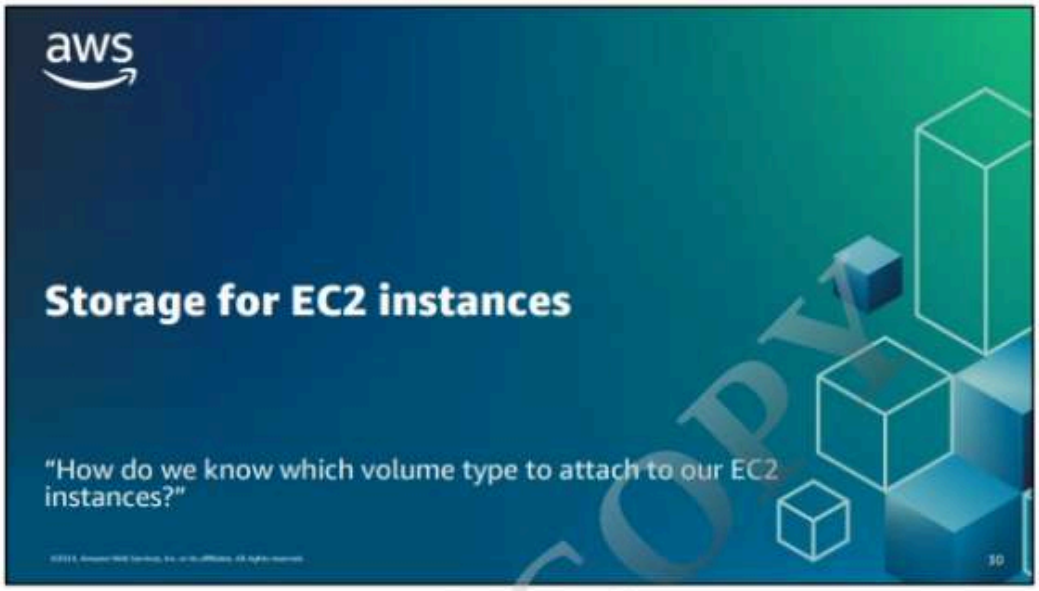
The diagram contains a user data script on an Amazon Linux 2 instance that runs at first launch. During instance launch, the EC2 instance requests a session token. It uses that token to request the public hostname of the EC2 instance at the following endpoint: http://169.254.169.254/latest/meta-data/public-hostname.

The instance metadata is then passed as the hostname in the operating system.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

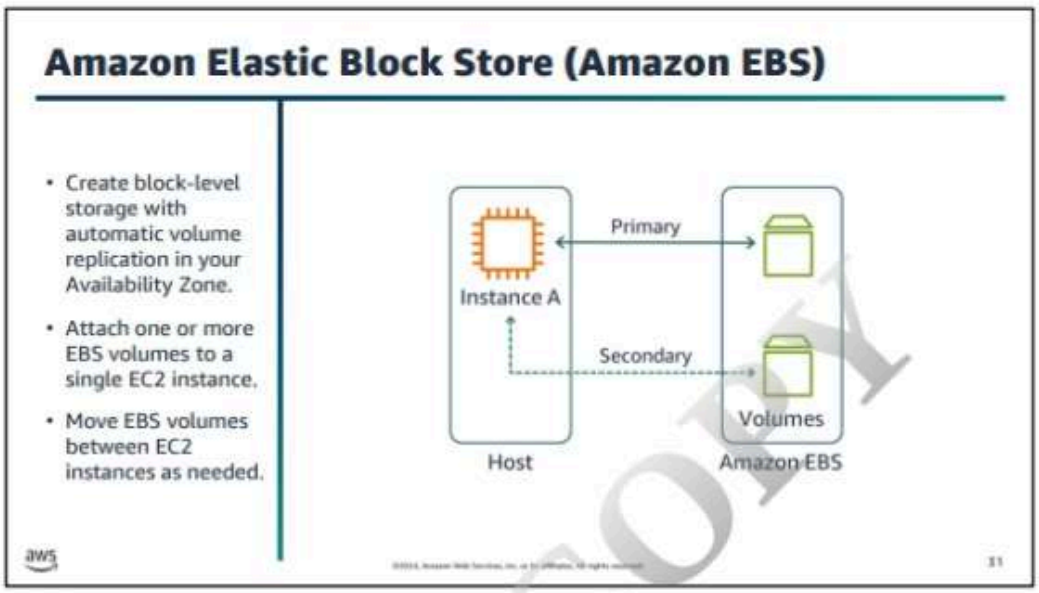


The compute operations manager asks, "How do we know which volume type to attach to our EC2 instances?" The team must learn more about Amazon EBS and instance store volume types.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Amazon EBS volumes provide durable, detachable, block-level storage for your Amazon EC2 instances. EBS volumes are mounted to the instances. Therefore, they can provide extremely low latency between where the data is stored and where it might be used on the instance. For this reason, you can use them to run a database with an Amazon EC2 instance.

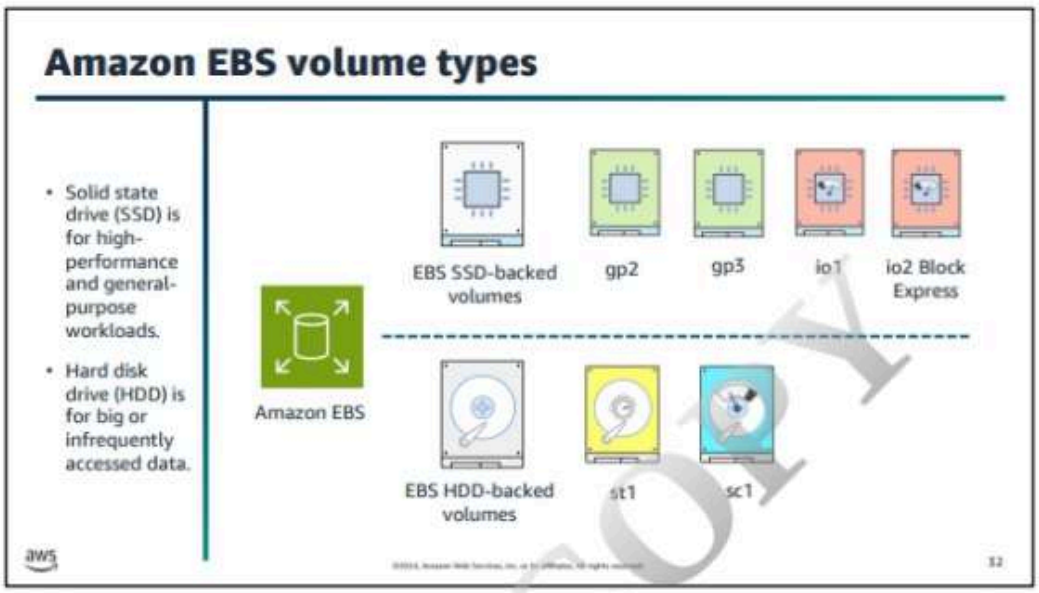
You can create Amazon EBS snapshots as a point-in-time copy of your data. You can also use them to store data for your AMIs. Snapshots are kept in Amazon Simple Storage Service (Amazon S3), and you can reuse them to create new EC2 instances later.

For more information about Amazon EBS, see "What is Amazon Elastic Block Store?" in the *Amazon EBS User Guide* at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/AmazonEBS.html.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



General purpose solid state drive (SSD) volumes (gp2 and gp3) offer cost-effective storage that is ideal for a broad range of use cases. These volume types are ideal for boot volumes, small and medium-sized databases, and development and test environments.

Provisioned input/output operations per second (IOPS) SSD volumes (io1 and io2 Block Express) are designed to meet the needs of I/O intensive workloads. For example, database workloads might be sensitive to storage performance and consistency. Provisioned IOPS SSD volumes use a consistent IOPS rate. You specify the rate when you create the volume. Amazon EBS delivers the provisioned performance 99.9 percent of the time.

Throughput-optimized hard disk drive (HDD) volumes (st1) provide low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. This volume type is a good fit for large, sequential workloads such as Amazon EMR; extract, transform, and load (ETL); data warehouses; and log processing.

Cold HDD (sc1) volumes provide low-cost magnetic storage that defines performance in terms of throughput rather than IOPS. sc1 is a good fit for large, sequential cold-data workloads. sc1 provides inexpensive block storage if you require infrequent access to your data.

For more information about Amazon EBS volume types, see "Amazon EBS volume types" in the *Amazon EBS User Guide* at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/ebs-volume-types.html.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Amazon EBS volume characteristics (1 of 2)

	General Purpose SSD		Provisioned IOPS SSD	
Volume type	gp2	gp3	io1	io2 Block Express
Description	Volume that balances price and performance for a wide variety of workloads		Highest performance volume for critical, IOPS-intensive, and throughput-intensive workloads that require low latency	
Size	1 gibibyte (GiB) to 16 TiB (tebibytes)		4 GiB to 16 TiB	4 GiB to 64 TiB
Maximum IOPS	16,000 (burst)	16,000 (no burst)	64,000	256,000
Maximum throughput per volume	250 mebibytes per second (MiBps)	1,000 MiBps	1,000 MiBps	4,000 MiBps

11

SSD-backed volumes are optimized for transactional workloads that involve frequent read/write operations with small I/O size, where the dominant performance attribute is IOPS.

Use cases for General Purpose SSD volumes include the following:
- Transactional workloads
- Virtual desktops
- Medium-sized, single-instance databases
- Low-latency interactive applications
- Boot volumes
- Development and test environments

Use cases for io1 Provisioned IOPS SSD volumes include the following:
- Workloads that require sustained IOPS performance or more than 16,000 IOPS
- I/O intensive database workloads

Specific use cases for io2 Block Express include the following:
- Sub-millisecond latency
- Sustained IOPS performance
- More than 64,000 IOPS or 1,000 MiBps of throughput

For more information about Amazon EBS SSD volumes, see "Solid state drive (SSD) volumes" in the *Amazon EBS User Guide* at https://docs.aws.amazon.com/ebs/latest/userguide/ebs-volume-types.html#vol-type-ssd.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Amazon EBS volume characteristics (2 of 2)

	Throughput Optimized HDD	Cold HDD
Volume type	st1	sc1
Description	Low-cost HDD volume designed for frequently accessed, throughput-intensive workloads	Lowest-cost HDD volume designed for less frequently accessed workloads
Size	125 GiB to 16 TiB	125 GiB to 16 TiB
Maximum IOPS	500	250
Maximum throughput per volume	500 MiBps	250 MiBps

34

HDD-backed volumes are optimized for large streaming workloads where throughput, measured in MiBps, is a better performance measure than IOPS.

Use cases for Throughput Optimized HDD volumes include the following:
- Big data
- Data warehouses
- Log processing

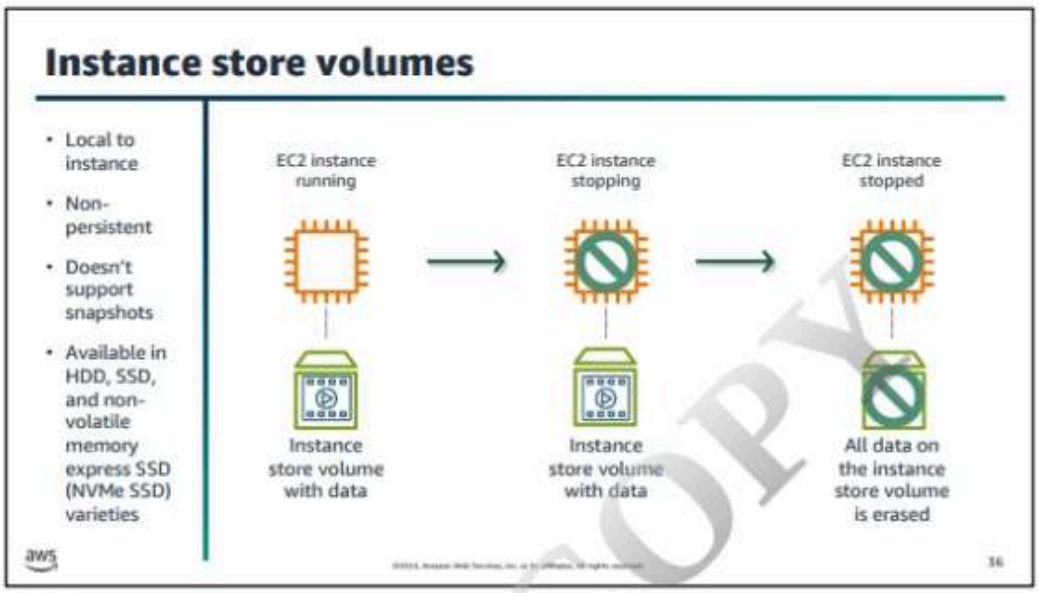
Use cases for Cold HDD volumes include the following:
- Throughput-oriented storage for data that is infrequently accessed
- Scenarios where the lowest storage cost is important

For more information about Amazon EBS HDD volumes, see "Hard disk drive (HDD) volumes" in the *Amazon EBS User Guide* at https://docs.aws.amazon.com/ebs/latest/userguide/ebs-volume-types.html#vol-type-hdd.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



An *instance store* provides temporary block-level storage for your instance. This storage is located on disks that are physically attached to the host computer.

An instance store is ideal for temporary storage of information that changes frequently, such as buffers, caches, scratch data, and other temporary content. It is also good for data that is replicated across a fleet of instances, such as a load-balanced pool of web servers.

An instance store has the following characteristics:
- Directly attached block-level storage
- Low-latency
- High IOPS and throughput
- Is reclaimed when the instance is stopped or terminated

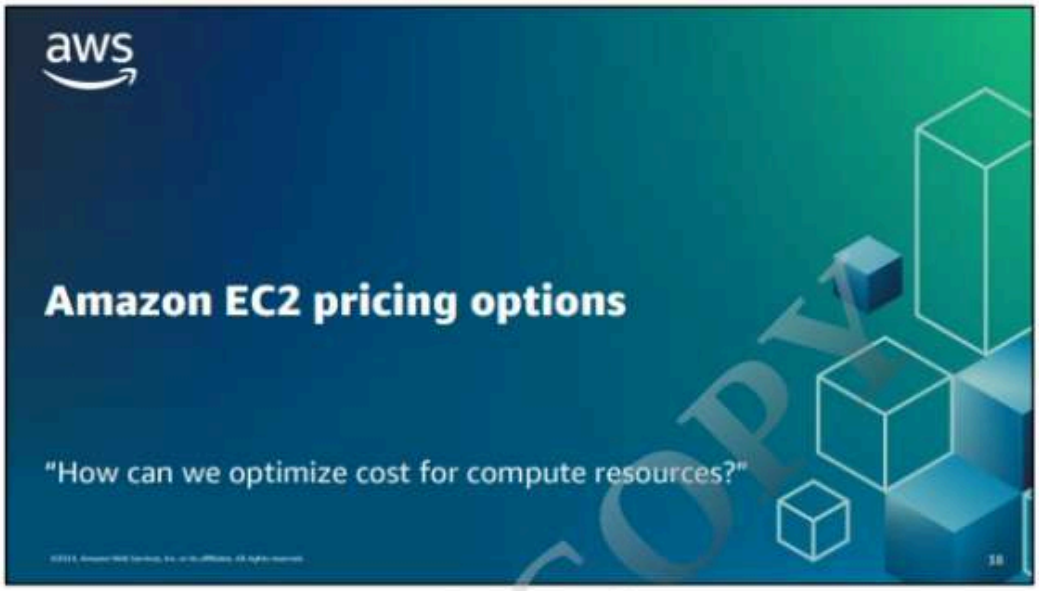
Instance stores are available in HDD, SSD, and non-volatile memory express SSD (NVMe SSD) varieties.

Use cases for an instance store include the following:
- Buffers
- Cache
- Temporary data

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

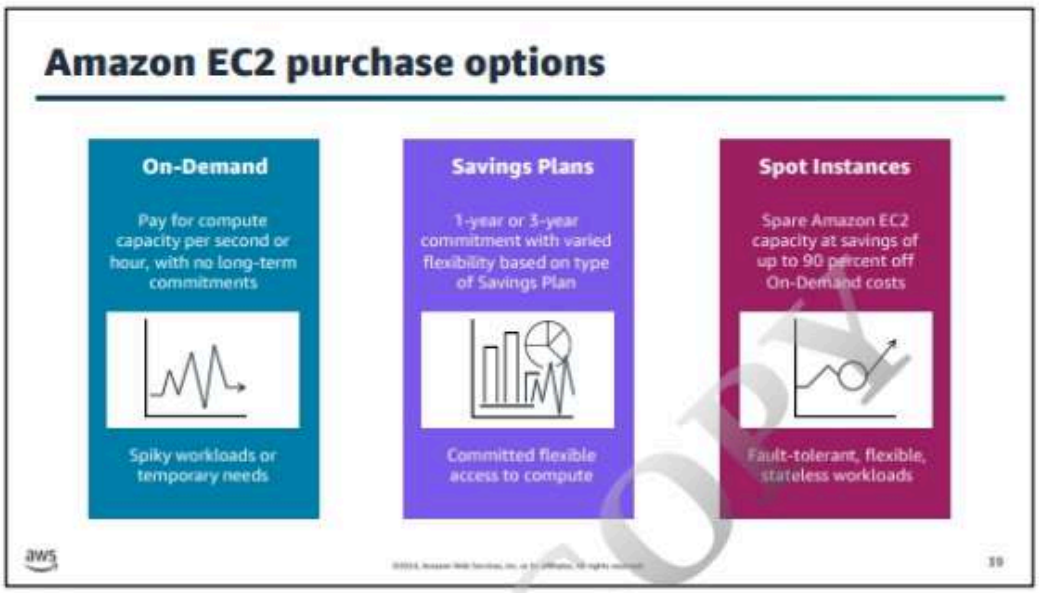


The compute operations manager asks, "How can we optimize cost for compute resources?" The team wants to know about different options for EC2 instance pricing models. You must explain how to balance commitment and flexibility with cost.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



With On-Demand Instances, you can pay for compute capacity by the second or hour, with no long-term commitments. On-Demand Instances are ideal for short-term, irregular workloads that cannot be interrupted. You can also choose On-Demand initially to determine your needs based on usage.

- Pay for compute capacity per second (for Linux, Windows, Windows with SQL Enterprise, Windows with SQL Standard, and Windows with SQL Web) or per hour (for all other OSs).
- There are no long-term commitments.
- There are no upfront payments.
- Increase or decrease your compute capacity depending on the demands of your application.

Savings Plans and Spot Instances give you cost savings with a different type of commitment. The next few slides describe when it is best to use which plans, based on your need.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

Savings Plan types

Compute Savings Plans

Greatest flexibility, up to **66 percent** off On-Demand rates, and applies to AWS Fargate and AWS Lambda usage

Flexible across:

- Instance family
- Size
- OS
- Tenancy
- Availability Zone
- Region

EC2 Instance Savings Plans

Provide the lowest prices, up to **72 percent** off On-Demand rates on the selected instance family in a specific AWS Region

Flexible across:

- Availability Zone
- Size
- OS
- Tenancy

40

Compute Savings Plans provide the most flexibility. The price can be up to 66 percent off On-Demand rates. These plans automatically apply to your EC2 instance usage, regardless of instance family, size, Region, Availability Zone, OS, or tenancy. They also apply to your Fargate and Lambda usage.

With a Compute Savings Plan, you can move a workload from C5 to M5, shift your usage from Ireland to London, or migrate your application from Amazon EC2 to Amazon ECS using Fargate at any time.

EC2 Instance Savings Plans provide savings up to 72 percent off On-Demand, with a commitment to a specific instance family in a chosen AWS Region. These plans automatically apply to usage within the specified family in a Region. They apply regardless of Availability Zone, size (for example, m5.xlarge or m5.2xlarge), OS (for example, Windows or Linux), or tenancy (Host, Dedicated, or Default).

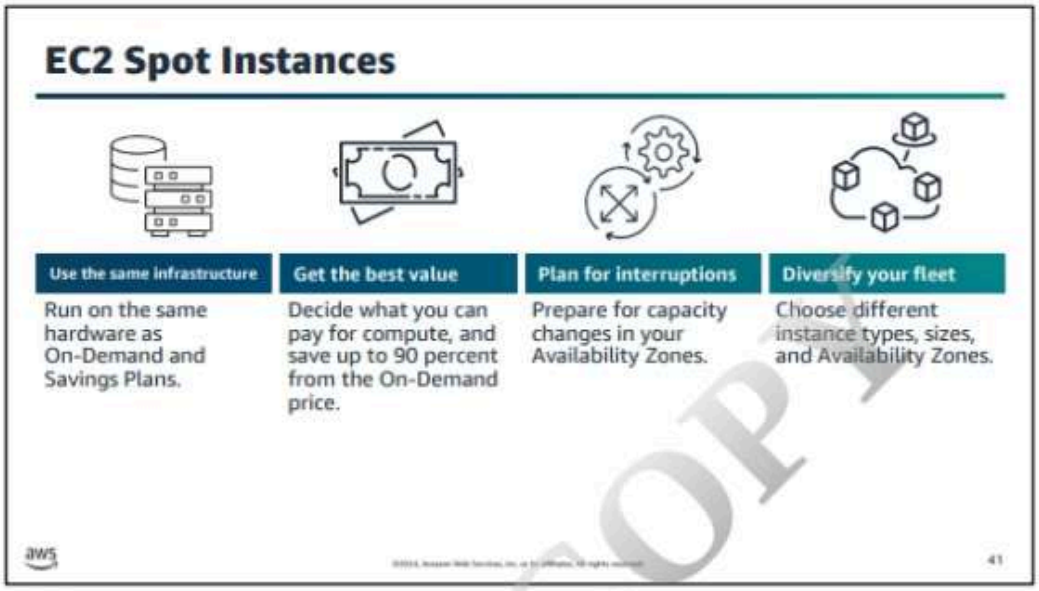
With an EC2 Instance Savings Plan, you can change your instance size within the instance family or the OS. You can also move from Dedicated tenancy to Default and continue to receive the discounted rate that your EC2 Instance Savings Plan provides.

We recommend Savings Plans over Reserved Instances. Like Reserved Instances, Savings Plans offer lower prices (up to 72 percent savings compared to On-Demand instance pricing). In addition, Savings Plans offer you the flexibility to change your usage as your needs evolve.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



A Spot Instance is an instance that uses spare EC2 host capacity. Spot Instances give you up to 90 percent savings compared to On-Demand Instances. Because Spot Instances permit you to request unused EC2 instances at steep discounts, you can lower your Amazon EC2 costs for flexible workloads.

The hourly price for a Spot Instance is called a Spot price. The Spot price of each instance type in each Availability Zone is set by Amazon EC2. The price is adjusted gradually, based on the long-term supply and demand for Spot Instances. Your Spot Instance runs whenever capacity is available, and if your requested maximum price is higher than the Spot price. If your Spot request cannot be fulfilled, it fails.

An interruption occurs when there is currently no capacity for your request at your maximum price. You receive a notification two minutes before the event.

You can get faster results by increasing throughput up to 10 times, while still staying under budget. You can still diversify instances by choosing different types, sizes, and Availability Zones.

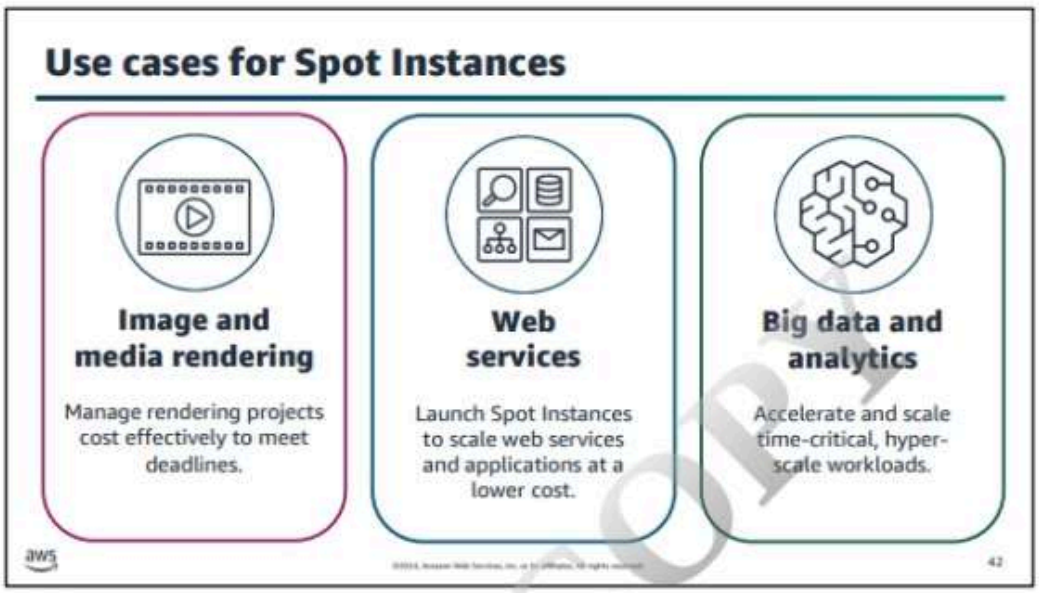
Launch your Spot Instances through AWS services, such as Amazon Elastic Container Service (Amazon ECS), AWS Batch, Amazon EMR, or by using integrated third parties.

For more information, see "Spot Instances" in the *Amazon Elastic Compute Cloud User Guide* at https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/using-spot-instances.html.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Spot Instances are ideal for your fault-tolerant, flexible, loosely coupled, or stateless workloads.

Some common use cases for Spot Instances include the following:

- **Image and media rendering** – You can manage and scale your on-premises or cloud rendering workloads cost effectively with near limitless capacity.
- **Web services** – Deploy an EC2 Spot Fleet behind your load balancer to scale to tens of thousands of instances, serving billions of service requests.
- **Big data and analytics** – Fast-track big data, machine learning, and natural language processing (NLP) workloads with Spot Instances. Spot Instances provide acceleration, scale, and deep cost savings to run time-critical, hyper-scale workloads for rapid data analysis.

Other use cases include:

- Containerized workloads
- Continuous integration/continuous delivery (CI/CD) and testing
- HPC

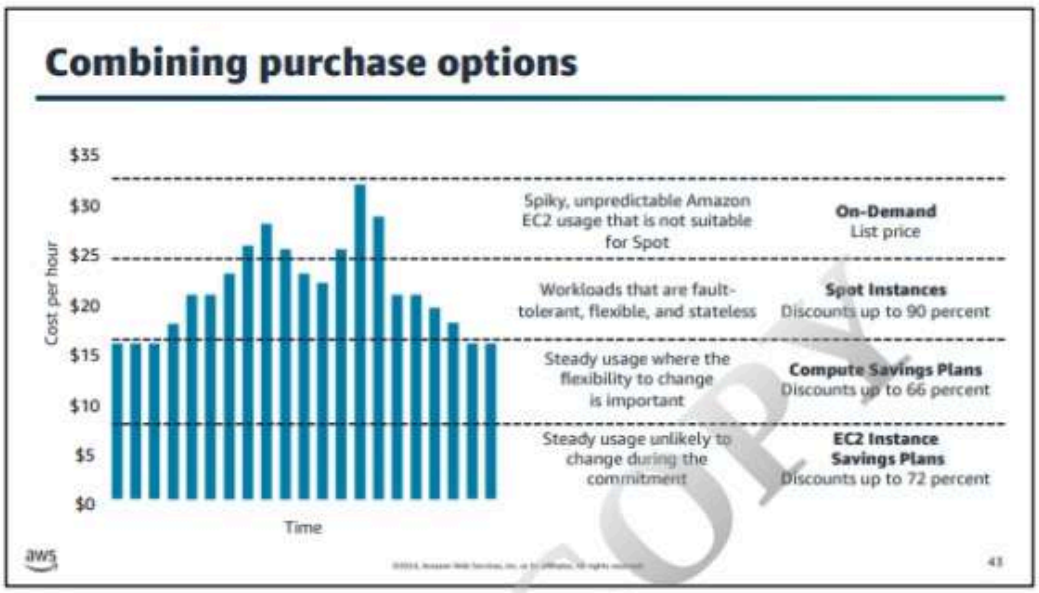
EC2 Spot Instances are also integrated into multiple AWS services, such as Amazon EC2 Auto Scaling groups, Amazon EMR, Amazon ECS, and AWS Batch.

Learn about how companies like Western Digital use Spot Instances. For more information, see "Western Digital HDD Simulation at Cloud Scale – 2.5 Million HPC Tasks, 40K EC2 Spot Instances" in the *AWS News Blog* at https://aws.amazon.com/blogs/aws/western-digital-hdd-simulation-at-cloud-scale-2-5-million-hpc-tasks-40k-ec2-spot-instances/.

DO NOT COPY javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



Create your strategy for purchasing EC2 instances to get the most for your budget.

- Use **Savings Plans** to budget for your defined compute needs. This part is more of a fixed cost, which will not change for you from month to month.
- Launch **Spot Instances** for your more flexible workloads that take into account failure or missing capacity short term.
- For the rest, use **On-Demand** EC2 instances and pay the list price for your compute.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

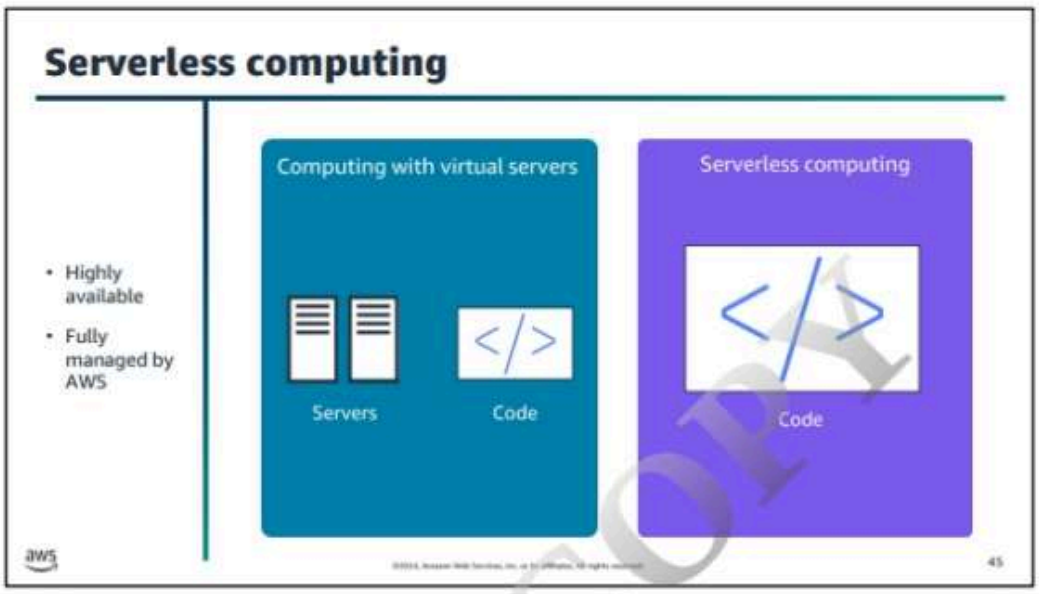
Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

The compute operations manager asks, "Where can we start with serverless compute options?" The team needs information about other hosting options in AWS. You can start by explaining more about AWS Lambda and how it can save on both compute cost and development resources.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



What is serverless computing?
With serverless computing, you can build and run applications and services without thinking about servers. Serverless applications don't require you to provision, scale, and manage any servers. You can build them for nearly any type of application or backend service. Everything that is required to run and scale your application with high availability is handled for you.

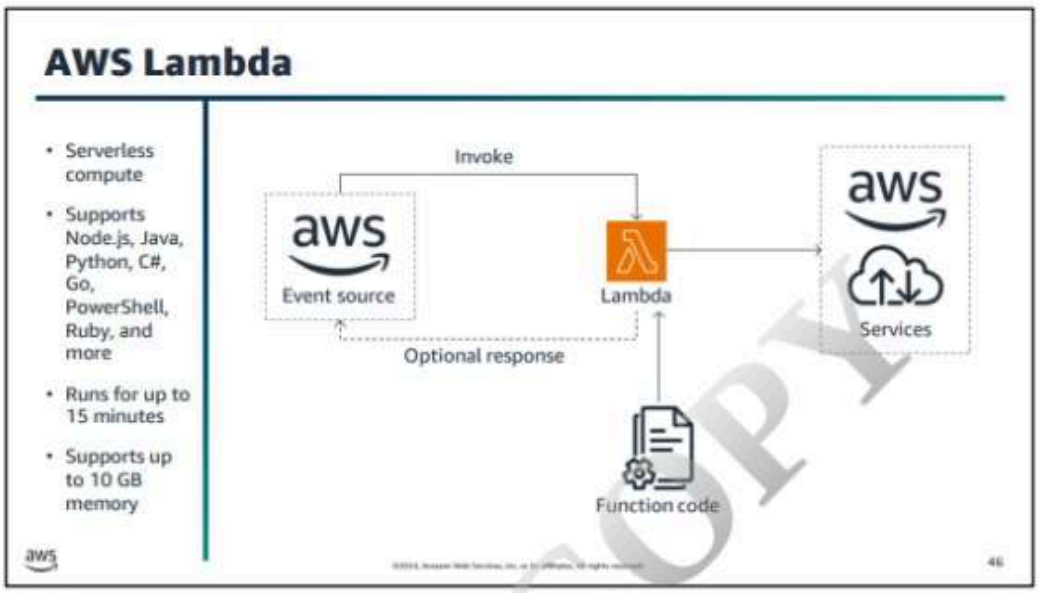
Why use serverless computing?
With serverless applications, your developers can focus on their core product instead of worrying about managing and operating servers or runtimes. Because of this reduced overhead, developers can reclaim time and energy that can be spent on developing great products that scale and are reliable.

For more information, see Serverless on AWS at https://aws.amazon.com/serverless/.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.



With Lambda, you can run code without provisioning or managing servers. The service runs your code on a high-availability compute infrastructure and performs all administration of the compute resources. These resources include the following:

- Server and OS maintenance
- Capacity provisioning and automatic scaling
- Code monitoring and logging

All that you must do is supply your code in one of the languages that Lambda supports: Node.js, Java, C#, Python, Go, Ruby, and PowerShell.

The core components of Lambda are the *event source* and the *Lambda function*. Event sources publish events. A Lambda function is the custom code that you write to process the events. Lambda runs your function.

A Lambda function consists of your code, associated dependencies, and configuration. Configuration includes information such as the following:

- The handler that will receive the event
- The AWS Identity and Access Management (IAM) role that Lambda can assume to run the Lambda function on your behalf
- The compute resource that you want allocated
- The delivery timeout

There is no additional charge for creating Lambda functions. You do incur charges for running a function and for data transfer between Lambda and other AWS services. Some optional Lambda features, for example, provisioned concurrency, also incur charges.

For more information about provisioned concurrency, see "Configuring reserved concurrency for a function" in the *AWS Lambda Developer Guide* at https://docs.aws.amazon.com/lambda/latest/dg/configuration-concurrency.html.

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.

Printed by: javierartiga.espublico@gmail.com. Printing is for personal, private use only. No part of this book may be reproduced or transmitted without publisher's prior permission. Violators will be prosecuted.

For more information about Lambda pricing, see AWS Lambda Pricing at https://aws.amazon.com/lambda/pricing/.

DO NOT COPY
javierartiga.espublico@gmail.com

© 2025, Amazon Web Services, Inc. or its affiliates. All rights reserved.