

PEARSON IT  
CERTIFICATION



Practice  
Tests



Flash  
Cards



Review  
Exercises



Study  
Planner

# Cert Guide

Advance your IT career with hands-on learning

# AWS Certified Solutions Architect – Associate

(SAA-C03)



MARK WILKINS

PEARSON IT  
CERTIFICATION



# Cert Guide

Advance your IT career with hands-on learning

# AWS Certified Solutions Architect – Associate

(SAA-C03)



Practice  
Tests



Flash  
Cards



Review  
Exercises



Study  
Planner



MARK WILKINS

## About This eBook

ePUB is an open, industry-standard format for eBooks. However, support of ePUB and its many features varies across reading devices and applications. Use your device or app settings to customize the presentation to your liking. Settings that you can customize often include font, font size, single or double column, landscape or portrait mode, and figures that you can click or tap to enlarge. For additional information about the settings and features on your reading device or app, visit the device manufacturer's Web site.

Many titles include programming code or configuration examples. To optimize the presentation of these elements, view the eBook in single-column, landscape mode and adjust the font size to the smallest setting. In addition to presenting code and configurations in the reflowable text format, we have included images of the code that mimic the presentation found in the print book; therefore, where the reflowable format may compromise the presentation of the code listing, you will see a “Click here to view code image” link. Click the link to view the print-fidelity code image. To return to the previous page viewed, click the Back button on your device or app.

# AWS Certified Solutions Architect – Associate (SAA-C03) Cert Guide

Mark Wilkins



# Pearson

## **AWS Certified Solutions Architect – Associate (SAA-C03) Cert Guide**

Copyright © 2023 by Pearson Education, Inc.

All rights reserved. No part of this book shall be reproduced, stored in a retrieval system, or transmitted by any means, electronic, mechanical, photocopying, recording, or otherwise, without written permission from the publisher. No patent liability is assumed with respect to the use of the information contained herein. Although every precaution has been taken in the preparation of this book, the publisher and author assume no responsibility for errors or omissions. Nor is any liability assumed for damages resulting from the use of the information contained herein.

ISBN-13: 978-0-13-794158-2

ISBN-10: 0-13-794158-7

Library of Congress Control Number: 2023930964

**ScoutAutomatedPrintCode**

Trademarks

All terms mentioned in this book that are known to be trademarks or service marks have been appropriately capitalized. Pearson IT Certification cannot attest to the accuracy of this information. Use of a term in this book should not be regarded as affecting the validity of any trademark or service mark.

### Warning and Disclaimer

Every effort has been made to make this book as complete and as accurate as possible, but no warranty or fitness is implied. The information provided is on an “as is” basis. The author and the publisher shall have neither liability nor responsibility to any person or entity with respect to any loss or damages arising from the information contained in this book.

### Special Sales

For information about buying this title in bulk quantities, or for special sales opportunities (which may include electronic versions; custom cover designs; and content particular to your business, training goals, marketing focus, or branding interests), please contact our corporate sales department at [corpsales@pearsoned.com](mailto:corpsales@pearsoned.com) or (800) 382-3419.

For government sales inquiries, please contact  
[governmentsales@pearsoned.com](mailto:governmentsales@pearsoned.com).

For questions about sales outside the U.S., please contact  
[intlcs@pearson.com](mailto:intlcs@pearson.com).

Vice President, IT Professional

Mark Taub

Director, ITP Product Management

Brett Bartow

Executive Editor

Nancy Davis

Development Editor

Christopher Cleveland

Managing Editor

Sandra Schroeder

Senior Project Editor

Tonya Simpson

Copy Editor

Bill McManus

Indexer

Jen Hinchliffe

Proofreader

Jen Hinchliffe

Technical Editor

Ralph Parisi

Publishing Coordinator

Cindy Teeters

Cover Designer

Chuti Prasertsith

Compositor

codeMantra

# **Pearson's Commitment to Diversity, Equity, and Inclusion**

Pearson is dedicated to creating bias-free content that reflects the diversity of all learners. We embrace the many dimensions of diversity, including but not limited to race, ethnicity, gender, socioeconomic status, ability, age, sexual orientation, and religious or political beliefs.

Education is a powerful force for equity and change in our world. It has the potential to deliver opportunities that improve lives and enable economic mobility. As we work with authors to create content for every product and service, we acknowledge our responsibility to demonstrate inclusivity and incorporate diverse scholarship so that everyone can achieve their potential through learning. As the world's leading learning company, we have a duty to help drive change and live up to our purpose to help more people create a better life for themselves and to create a better world.

Our ambition is to purposefully contribute to a world where

- Everyone has an equitable and lifelong opportunity to succeed through learning

- Our educational products and services are inclusive and represent the rich diversity of learners
- Our educational content accurately reflects the histories and experiences of the learners we serve
- Our educational content prompts deeper discussions with learners and motivates them to expand their own learning (and worldview)

While we work hard to present unbiased content, we want to hear from you about any concerns or needs with this Pearson product so that we can investigate and address them.

Please contact us with concerns about any potential bias at  
<https://www.pearson.com/report-bias.html>.

# Contents at a Glance

Introduction

Chapter 1 Understanding the Foundations of AWS Architecture

Chapter 2 The AWS Well-Architected Framework

Chapter 3 Designing Secure Access to AWS Resources

Chapter 4 Designing Secure Workloads and Applications

Chapter 5 Determining Appropriate Data Security Controls

Chapter 6 Designing Resilient Architecture

Chapter 7 Designing Highly Available and Fault-Tolerant Architecture

Chapter 8 High-Performing and Scalable Storage Solutions

Chapter 9 Designing High-Performing and Elastic Compute Solutions

Chapter 10 Determining High-Performing Database Solutions

Chapter 11 High-Performing and Scalable Networking Architecture

Chapter 12 Designing Cost-Optimized Storage Solutions

Chapter 13 Designing Cost-Effective Compute Solutions

Chapter 14 Designing Cost-Effective Database Solutions

Chapter 15 Designing Cost-Effective Network Architectures

Chapter 16 Final Preparation

[Appendix A Answers to the “Do I Know This Already?”](#)

[Quizzes and Q&A Sections](#)

[Appendix B AWS Certified Solutions Architect – Associate](#)

[\(SAA-C03\) Cert Guide Exam Updates](#)

[Glossary of Key Terms](#)

[Index](#)

**Online Elements:**

[Appendix C Study Planner](#)

[Glossary of Key Terms](#)

# Table of Contents

Introduction

Chapter 1 Understanding the Foundations of AWS Architecture

Essential Characteristics of AWS Cloud Computing

AWS Cloud Computing and NIST

On-Demand Self-Service

Broad Network Access

Resource Pooling

Rapid Elasticity

Measured Service

Moving to AWS

Infrastructure as a Service (IaaS)

Platform as a Service (PaaS)

Operational Benefits of AWS

Cloud Provider Responsibilities

Security at AWS

Network Security at AWS

Application Security at AWS

Migrating Applications

Applications That Can Be Moved to AWS and Hosted

on an EC2 Instance with No Changes

Applications with Many Local Dependencies That Cause Problems When Being Moved to the Cloud  
Replacing an Existing Application with a SaaS Application Hosted by a Public Cloud Provider  
Applications That Should Remain On Premises and Eventually Be Deprecated

The AWS Well-Architected Framework

The Well-Architected Tool

AWS Services Cheat Sheet  
In Conclusion

Chapter 2 The AWS Well-Architected Framework

“Do I Know This Already?”

Foundation Topics

The Well-Architected Framework

Operational Excellence Pillar

Security Pillar

*Defense in Depth*

Reliability Pillar

Performance Efficiency Pillar

Cost Optimization Pillar

Sustainability Pillar

Designing a Workload SLA

Reliability and Performance Are Linked

Disaster Recovery

## Placing Cloud Services

*Data Residency and Compute Locations*

*Caching Data with CDNs*

*Data Replication*

*Load Balancing Within and Between Regions*

*Failover Architecture*

## Deployment Methodologies

*Factor 1: Use One Codebase That Is Tracked with Version Control to Allow Many Deployments*

*AWS CodeCommit*

*Factor 2: Explicitly Declare and Isolate Dependencies*

*Factor 3: Store Configuration in the Environment*

*Factor 4: Treat Backing Services as Attached Resources*

*Factor 5: Separate Build and Run Stages*

*Factor 6: Execute an App as One or More Stateless Processes*

*Factor 7: Export Services via Port Binding*

*Factor 8: Scale Out via the Process Model*

*Factor 9: Maximize Robustness with Fast Startup and Graceful Shutdown*

*Factor 10: Keep Development, Staging, and Production as Similar as Possible*

*Factor 11: Treat Logs as Event Streams*

## Factor 12: Run Admin/Management Tasks as One-Off Processes

Exam Preparation Tasks

Review All Key Topics

Define Key Terms

Q&A

## Chapter 3 Designing Secure Access to AWS Resources

“Do I Know This Already?”

Foundation Topics

Identity and Access Management (IAM).

IAM Policy Definitions

IAM Authentication

Requesting Access to AWS Resources

The Authorization Process

Actions

IAM Users and Groups

The Root User

The IAM User

Creating an IAM User

IAM User Access Keys

IAM Groups

Signing In as an IAM User

IAM Account Details

Creating a Password Policy

[Rotating Access Keys](#)

[Using Multi-Factor Authentication](#)

[Creating IAM Policies](#)

[IAM Policy Types](#)

[Identity-Based Policies](#)

[Resource-Based Policies](#)

[Inline Policies](#)

[IAM Policy Creation](#)

[Policy Elements](#)

[Reading a Simple JSON Policy](#)

[Policy Actions](#)

[Additional Policy Control Options](#)

[Reviewing Policy Permissions](#)

[IAM Policy Versions](#)

[Using Conditional Elements](#)

[Using Tags with IAM Identities](#)

[IAM Roles](#)

[When to Use IAM Roles](#)

[AWS Services Perform Actions on Your Behalf](#)

[EC2 Instances Hosting Applications Need Access to AWS Resources](#)

[Access to AWS Accounts by Third Parties](#)

[Web Identity Federation](#)

[SAML 2.0 Federation](#)

Cross-Account Access

AWS Security Token Service

IAM Best Practices

IAM Security Tools

IAM Cheat Sheet

AWS Identity Center

AWS Organizations

AWS Organizations Cheat Sheet

AWS Resource Access Manager

AWS Control Tower

Exam Preparation Tasks

Review All Key Topics

Define Key Terms

Q&A

Chapter 4 Designing Secure Workloads and Applications

“Do I Know This Already?”

Foundation Topics

Securing Network Infrastructure

Networking Services Located at Edge Locations

AWS Shield (Standard and Advanced)

AWS Web Application Firewall (WAF)

VPC Networking Services for Securing Workloads

Route Tables

The Main Route Table

[Security Groups](#)

[Security Groups Cheat Sheet](#)

[Web Server Inbound Ports](#)

[Database Server Inbound Ports](#)

[Administration Access](#)

[Understanding Ephemeral Ports](#)

[Security Group Planning](#)

[Network ACLs](#)

[Network ACL Implementation Details](#)

[Network ACL Cheat Sheet](#)

[Network ACL Rule Processing](#)

[VPC Flow Logs](#)

[NAT Services](#)

[NAT Gateway Service](#)

[NAT Instance](#)

[AWS NAT Gateway Service Cheat Sheet](#)

[Amazon Cognito](#)

[User Pool](#)

[Federated Identity Provider](#)

[External Connections](#)

[Virtual Private Gateway](#)

[Customer Gateway](#)

[AWS Managed VPN Connection Options](#)

[Understanding Route Propagation](#)

[AWS Direct Connect](#)

[AWS Direct Connect Gateway](#)

[AWS Direct Connect Cheat Sheet](#)

[Amazon GuardDuty](#)

[Amazon GuardDuty Cheat Sheet](#)

[Amazon Macie](#)

[Amazon Macie Cheat Sheet](#)

[Security Services for Securing Workloads](#)

[AWS CloudTrail](#)

[Creating an AWS CloudWatch Trail](#)

[AWS CloudTrail Cheat Sheet](#)

[AWS Secrets Manager](#)

[Amazon Inspector](#)

[AWS Trusted Advisor](#)

[AWS Config](#)

[Exam Preparation Tasks](#)

[Review All Key Topics](#)

[Define Key Terms](#)

[Q&A](#)

[Chapter 5 Determining Appropriate Data Security Controls](#)

[“Do I Know This Already?”](#)

[Foundation Topics](#)

[Data Access and Governance](#)

[Data Retention and Classification](#)

Infrastructure Security  
IAM Controls  
Detective Controls  
Amazon EBS Encryption  
Amazon S3 Bucket Security  
S3 Storage at Rest  
Amazon S3 Object Lock Policies  
Legal Hold  
Amazon S3 Glacier Storage at Rest  
Data Backup and Replication  
AWS Key Management Service  
Envelope Encryption  
AWS KMS Cheat Sheet  
AWS CloudHSM  
AWS Certificate Manager  
Encryption in Transit  
Exam Preparation Tasks  
Review All Key Topics  
Define Key Terms  
Q&A  
Chapter 6 Designing Resilient Architecture  
“Do I Know This Already?”  
Foundation Topics  
Scalable and Resilient Architecture

[Scalable Delivery from Edge Locations](#)  
[Stateful Versus Stateless Application Design](#)  
[Changing User State Location](#)  
[User Session Management](#)  
[Container Orchestration](#)  
[Migrating Applications to Containers](#)  
[Resilient Storage Options](#)  
[Application Integration Services](#)  
[Amazon Simple Notification Service](#)  
[\*Amazon SNS Cheat Sheet\*](#)  
[Amazon Simple Queue Service](#)  
[\*SQS Components\*](#)  
[\*Amazon SQS Cheat Sheet\*](#)  
[AWS Step Functions](#)  
[Amazon EventBridge](#)  
[Amazon API Gateway](#)  
[\*API Gateway Cheat Sheet\*](#)  
[Building a Serverless Web App](#)  
[\*Step 1: Create a Static Website\*](#)  
[\*Step 2: User Authentication\*](#)  
[\*Step 3: Create the Serverless Backend Components\*](#)  
[\*Step 4: Set Up the API Gateway\*](#)  
[\*Step 5: Register for the Conference\*](#)

## Automating AWS Infrastructure

### AWS CloudFormation

*CloudFormation Components*

*CloudFormation Templates*

*CloudFormation Stacks*

*CloudFormation Stack Sets*

*Third-Party Solutions*

### AWS Service Catalog

## AWS Elastic Beanstalk

*Updating Elastic Beanstalk Applications*

## Exam Preparation Tasks

*Review All Key Topics*

*Define Key Terms*

*Q&A*

## Chapter 7 Designing Highly Available and Fault-Tolerant Architecture

*“Do I Know This Already?”*

*Foundation Topics*

*High Availability and Fault Tolerance*

*High Availability in the Cloud*

*Reliability*

*AWS Regions and Availability Zones*

*Availability Zones*

*Availability Zone Distribution*

[Planning Network Topology](#)

[Local Zones](#)

[Wavelength Zones](#)

[AWS Services Use Cases](#)

[Choosing an AWS Region](#)

[Compliance Rules](#)

[Understanding Compliance Rules at AWS: Use Case](#)

[AWS Compliance Standards](#)

[HIPAA](#)

[NIST](#)

[AWS GovCloud](#)

[Latency Concerns](#)

[Services Offered in Each AWS Region](#)

[Calculating Costs](#)

[Distributed Design Patterns](#)

[Designing for High Availability and Fault Tolerance](#)

[Removing Single Points of Failure](#)

[Immutable Infrastructure](#)

[Storage Options and Characteristics](#)

[Failover Strategies](#)

[Backup and Restore](#)

[Pilot Light](#)

[Warm Standby](#)

## Multi-Region Scenarios

Warm Standby with Amazon Aurora

Active-Active

Single and Multi-Region Recovery Cheat Sheet

Disaster Recovery Cheat Sheet

## AWS Service Quotas

AWS Service Quotas Cheat Sheet

## Amazon Route 53

Route 53 Health Checks

Route 53 Routing Policies

Route 53 Traffic Flow Policies

Alias Records

Route 53 Resolver

## Exam Preparation Tasks

Review All Key Topics

Define Key Terms

Q&A

## Chapter 8 High-Performing and Scalable Storage Solutions

“Do I Know This Already?”

Foundation Topics

AWS Storage Options

Workload Storage Requirements

Amazon Elastic Block Store

EBS Volume Types

[General Purpose SSD \(gp2/gp3\)](#)

[Elastic EBS Volumes](#)

[Attaching an EBS Volume](#)

[Amazon EBS Cheat Sheet](#)

[EBS Snapshots](#)

[Taking a Snapshot from a Linux Instance](#)

[Taking a Snapshot from a Windows Instance](#)

[Fast Snapshot Restore](#)

[Snapshot Administration](#)

[EBS Recycle Bin](#)

[Snapshot Cheat Sheet](#)

[Local EC2 Instance Storage Volumes](#)

[Amazon Elastic File System](#)

[EFS Performance Modes](#)

[EFS Throughput Modes](#)

[EFS Security](#)

[EFS Storage Classes](#)

[EFS Lifecycle Management](#)

[Amazon EFS Cheat Sheet](#)

[AWS DataSync](#)

[Amazon FSx for Windows File Server](#)

[Amazon FSx for Windows File Server Cheat Sheet](#)

[Amazon Simple Storage Service](#)

[Amazon S3 Bucket Concepts](#)

[Amazon S3 Data Consistency](#)

[Amazon S3 Storage Classes](#)

[Amazon S3 Management](#)

[S3 Bucket Versioning](#)

[Amazon S3 Access Points](#)

[Multi-Region Access Points](#)

[Preselected URLs for S3 Objects](#)

[S3 Cheat Sheet](#)

[Amazon S3 Glacier](#)

[Vaults and Archives](#)

[S3 Glacier Retrieval Policies](#)

[S3 Glacier Deep Archive](#)

[Amazon S3 Glacier Cheat Sheet](#)

[AWS Data Lake](#)

[AWS Lake Formation](#)

[Structured and Unstructured Data](#)

[Analytical Tools and Datasets](#)

[AWS Glue](#)

[Analytic Services](#)

[Amazon Kinesis Data Streams](#)

[Exam Preparation Tasks](#)

[Review All Key Topics](#)

[Define Key Terms](#)

[Q&A](#)

## Chapter 9 Designing High-Performing and Elastic Compute Solutions

“Do I Know This Already?”

Foundation Topics

AWS Compute Services

AWS EC2 Instances

Amazon Machine Images

AWS AMIs

Creating a Custom AMI

AMI Build Considerations

Amazon EC2 Image Builder

AWS Lambda

AWS Lambda Integration

Lambda Settings

AWS Lambda Cheat Sheet

Amazon Container Services

Amazon Elastic Container Service

AWS ECS Task Definition Choices

Amazon Elastic Kubernetes Service

Monitoring with AWS CloudWatch

CloudWatch Basic Monitoring

CloudWatch Logs

Collecting Data with the CloudWatch Agent

Planning for Monitoring

[Amazon CloudWatch Integration](#)  
[Amazon CloudWatch Terminology](#)  
[Creating a CloudWatch Alarm](#)  
[Additional Alarm and Action Settings](#)  
[Amazon CloudWatch Cheat Sheet](#)  
[Auto Scaling Options at AWS](#)  
[EC2 Auto Scaling](#)  
[EC2 Auto Scaling Operation](#)  
[Launch Configuration](#)  
[Launch Templates](#)  
[Auto Scaling Groups](#)  
[Scaling Options for Auto Scaling Groups](#)  
[Management Options for Auto Scaling Groups](#)  
[Cooldown Period](#)  
[Termination Policy](#)  
[Lifecycle Hooks](#)  
[EC2 Auto Scaling Cheat Sheet](#)  
[AWS Auto Scaling](#)  
[Exam Preparation Tasks](#)  
[Review All Key Topics](#)  
[Define Key Terms](#)  
[Q&A](#)  
[Chapter 10 Determining High-Performing Database Solutions](#)  
[“Do I Know This Already?”](#)

[Foundation Topics](#)

[AWS Cloud Databases](#)

[Amazon Relational Database Service](#)

[Amazon RDS Database Instances](#)

[Database Instance Class Types](#)

[High-Availability Design for RDS](#)

[Multi-AZ RDS Deployments](#)

[Big-Picture RDS Installation Steps](#)

[Monitoring Database Performance](#)

[Best Practices for RDS](#)

[Amazon Relational Database Service Proxy](#)

[Amazon RDS Cheat Sheet](#)

[Amazon Aurora](#)

[Amazon Aurora Storage](#)

[Amazon Aurora Replication](#)

[Communicating with Amazon Aurora](#)

[Amazon Aurora Cheat Sheet](#)

[Amazon DynamoDB](#)

[Amazon DynamoDB Tables](#)

[Provisioning Table Capacity](#)

[Adaptive Capacity](#)

[Data Consistency](#)

[ACID and Amazon DynamoDB](#)

[Global Tables](#)

[Amazon DynamoDB Accelerator](#)

[Backup and Restoration](#)

[Amazon DynamoDB Cheat Sheet](#)

[Amazon ElastiCache](#)

[Amazon ElastiCache for Memcached](#)

[Amazon ElastiCache for Memcached Cheat Sheet](#)

[Amazon ElastiCache for Redis](#)

[Amazon ElastiCache for Redis Cheat Sheet](#)

[ElastiCache for Redis: Global Datastore](#)

[Amazon Redshift](#)

[Amazon Redshift Cheat Sheet](#)

[Exam Preparation Tasks](#)

[Review All Key Topics](#)

[Define Key Terms](#)

[Q&A](#)

[Chapter 11 High-Performing and Scalable Networking](#)

[Architecture](#)

[“Do I Know This Already?”](#)

[Foundation Topics](#)

[Amazon CloudFront](#)

[How Amazon CloudFront Works](#)

[Regional Edge Caches](#)

[CloudFront Use Cases](#)

[HTTPS Access](#)

[Serving Private Content](#)

[Using Signed URLs](#)

[Using an Origin Access Identifier](#)

[Restricting Distribution of Content](#)

[CloudFront Origin Failover](#)

[Video-on-Demand and Live Streaming Support](#)

[Edge Functions](#)

[CloudFront Functions](#)

[Lambda@Edge Functions](#)

[Lambda@Edge Use Cases](#)

[CloudFront Cheat Sheet](#)

[AWS Global Accelerator](#)

[Elastic Load Balancing Service](#)

[Application Load Balancer Features](#)

[Application Load Balancer Deployment](#)

[Health Checks](#)

[Target Group Attributes](#)

[Sticky Session Support](#)

[Access Logs](#)

[ALB Cheat Sheet](#)

[Network Load Balancer](#)

[NLB Cheat Sheet](#)

[Multi-Region Failover](#)

[CloudWatch Metrics](#)

## AWS VPC Networking

The Shared Security Model

AWS Networking Terminology

VPC Cheat Sheet

Creating a VPC

Using the Create VPC Wizard

Using the AWS CLI to Create a VPC

How Many VPCs Does Your Organization Need?

Creating the VPC CIDR Block

## Subnets

Subnet Cheat Sheet

## IP Address Types

Private IPv4 Addresses

Private IPv4 Address Summary

Public IPv4 Addresses

Elastic IP Addresses

Public IPv4 Address Cheat Sheet

Inbound and Outbound Traffic Charges

## Bring-Your-Own IP

The BYOIP Process

IPv6 Addresses

VPC Flow Logs

## Connectivity Options

VPC Peering

## Establishing a Peering Connection

### VPC Endpoints

*VPC Gateway Endpoints*

*VPC Interface Endpoints*

*Endpoint Services*

Exam Preparation Tasks

Review All Key Topics

Define Key Terms

Q&A

## Chapter 12 Designing Cost-Optimized Storage Solutions

“Do I Know This Already?”

Foundation Topics

Calculating AWS Costs

*Cloud Service Costs*

*Tiered Pricing at AWS*

*Management Tool Pricing Example: AWS Config*

*AWS Config Results*

Cost Management Tools

*AWS Cost Explorer*

*AWS Budgets*

*AWS Cost and Usage Reports*

*Managing Costs Cheat Sheet*

*Tagging AWS Resources*

*Using Cost Allocation Tags*

[Storage Types and Costs](#)

[AWS Backup](#)

[Lifecycle Rules](#)

[AWS Backup Cheat Sheet](#)

[Data Transfer Costs](#)

[AWS Storage Gateway](#)

[AWS Storage Gateway Cheat Sheet](#)

[Exam Preparation Tasks](#)

[Review All Key Topics](#)

[Define Key Terms](#)

[Q&A](#)

[Chapter 13 Designing Cost-Effective Compute Solutions](#)

[“Do I Know This Already?”](#)

[Foundation Topics](#)

[EC2 Instance Types](#)

[What Is a vCPU?](#)

[EC2 Instance Choices](#)

[Dedicated Host](#)

[Dedicated Hosts Cheat Sheet](#)

[Dedicated Instances](#)

[Placement Groups](#)

[EC2 Instance Purchasing Options](#)

[EC2 Pricing—On-demand](#)

[On-demand Instance Service Quotas](#)

Reserved Instances  
Term Commitment  
Payment Options  
EC2 Reserved Instance Types  
Scheduled Reserved EC2 Instances  
Regional and Zonal Reserved Instances  
Savings Plans  
Spot Instances  
Spot Fleet Optimization Strategies  
Spot Capacity Pools  
EC2 Pricing Cheat Sheet  
Compute Tools and Utilities  
Strategies for Optimizing Compute  
Matching Compute Utilization with Requirements  
Compute Scaling Strategies  
Exam Preparation Tasks  
Review All Key Topics  
Define Key Terms  
Q&A

Chapter 14 Designing Cost-Effective Database Solutions

“Do I Know This Already?”  
Foundation Topics  
Database Design Choices  
RDS Deployments

[RDS Costs Cheat Sheet](#)

[RDS Database Design Solutions](#)

[NoSQL Deployments](#)

[NoSQL Costs Cheat Sheet](#)

[Migrating Databases](#)

[AWS Schema Conversion Tool](#)

[Database Data Transfer Costs](#)

[Data Transfer Costs and RDS](#)

[Data Transfer Costs with DynamoDB](#)

[Data Transfer Costs with Amazon Redshift](#)

[Data Transfer Costs with DocumentDB](#)

[Data Transfer Costs Cheat Sheet](#)

[Database Retention Policies](#)

[Database Backup Policies Cheat Sheet](#)

[Exam Preparation Tasks](#)

[Review All Key Topics](#)

[Define Key Terms](#)

[Q&A](#)

[Chapter 15 Designing Cost-Effective Network Architectures](#)

[“Do I Know This Already?”](#)

[Foundation Topics](#)

[Networking Services and Connectivity Costs](#)

[Elastic Load Balancing Deployments](#)

[NAT Devices](#)

[AWS CloudFront](#)

[CloudFront Pricing Cheat Sheet](#)

[VPC Endpoints](#)

[Network Services from On-Premises Locations](#)

[Data Transfer Costs](#)

[Accessing AWS Services in the Same Region](#)

[Workload Components in the Same Region](#)

[Accessing AWS Services in Different Regions](#)

[Data Transfer at Edge Locations](#)

[Network Data Transfer](#)

[Public Versus Private Traffic Charges](#)

[Data Transfer Costs Cheat Sheet](#)

[Exam Preparation Tasks](#)

[Review All Key Topics](#)

[Define Key Terms](#)

[Q&A](#)

[Chapter 16 Final Preparation](#)

[Exam Information](#)

[Tips for Getting Ready for the Exam](#)

[Scheduling Your Exam](#)

[Tools for Final Preparation](#)

[Pearson Test Prep Practice Test Software and](#)

[Questions on the Website](#)

[Accessing the Pearson Test Prep Software Online](#)

*Accessing the Pearson Test Prep Software Offline*

*Customizing Your Exams*

*Updating Your Exams*

*Premium Edition*

*Chapter-Ending Review Tools*

*Suggested Plan for Final Review/Study*

*Summary*

*Appendix A Answers to the “Do I Know This Already?”*

*Quizzes and Q&A Sections*

*Appendix B AWS Certified Solutions Architect – Associate*

*(SAA-C03) Cert Guide Exam Updates*

*Glossary of Key Terms*

*Index*

**Online Elements:**

*Appendix C Study Planner*

*Glossary of Key Terms*

## About the Author

**Mark Wilkins** is an electronics engineering technologist with a wealth of experience in designing, deploying, and supporting software and hardware technology in the corporate and small business world. Since 2013, Mark has focused on supporting and designing cloud service solutions with Amazon Web Services, Microsoft Azure, and the IBM Cloud. He is certified as an AWS Certified Solutions Architect – Associate. Mark is also a Microsoft Certified Trainer (MCT) and holds certifications in MCTS, MCSA, Server Virtualization with Windows Server Hyper-V, and Azure Cloud Services.

Mark worked as a technical evangelist for IBM SoftLayer from 2013 through 2016 and taught both SoftLayer fundamentals and SoftLayer design classes to many Fortune 500 companies in Canada, the United States, Europe, and Australia. As former course director for Global Knowledge, Mark developed and taught many technical seminars, including Configuring Active Directory Services, Configuring Group Policy, and Cloud and Virtualization Essentials. Mark currently develops AWS curriculum on AWS cloud services and certification for O'Reilly Media and LinkedIn Learning. To learn more about what Mark finds interesting about the cloud, visit The Cloud Thingy, at <https://thecloudthingy.substack.com/>. To learn more about the

AWS cloud and AWS certification, check out Mark's YouTube channel at <http://www.youtube.com/@SAA-C03>.

Mark's published books include *Windows 2003 Registry for Dummies*, *Administering SMS 3.0*, *Administering Active Directory*, and *Learning Amazon Web Services (AWS): A Hands-On Guide to the Fundamentals of AWS Cloud*.

## Dedication

*I would like to dedicate this book to my grandson, Silas, a future nerd. And to Bruce, one of our cats, for making me take breaks when he wanted.*

## Acknowledgments

This manuscript was made truly great by the incredible project management of Tonya Simpson, who went above and beyond! Thanks so much.

I would also like to express my gratitude to Chris Cleveland, the development editor of this book. I was lucky to work with him on this text. Chris helped make this book several cuts above the rest.

Finally, thanks so much to Nancy Davis, my tireless acquisitions editor. Nancy very patiently made this book a reality.

## About the Technical Reviewer

**Ralph Parisi** is a certified Champion Authorized Amazon instructor and has been teaching AWS courses for 6 years. Ralph has been an instructor for more than 20 years and has taught technical classes for Microsoft Exchange Server, Microsoft Windows Server, Active Directory, Group Policy, Citrix XenDesktop, and XenApp. Ralph has worked as a consultant to large corporations architecting Exchange Server and Active Directory solutions and migrations. Ralph has also worked with various companies as a technical writer. Ralph lives in North Carolina with his wife and Saluki, Dillon.

## We Want to Hear from You!

As the reader of this book, *you* are our most important critic and commentator. We value your opinion and want to know what we're doing right, what we could do better, what areas you'd like to see us publish in, and any other words of wisdom you're willing to pass our way.

We welcome your comments. You can email or write to let us know what you did or didn't like about this book—as well as what we can do to make our books better.

*Please note that we cannot help you with technical problems related to the topic of this book.*

When you write, please be sure to include this book's title and author as well as your name and email address. We will carefully review your comments and share them with the author and editors who worked on the book.

Email: [community@informit.com](mailto:community@informit.com)

## Reader Services

Register your copy of *AWS Certified Solutions Architect – Associate (SAA-C03) Cert Guide* at [www.pearsonitcertification.com](http://www.pearsonitcertification.com) for convenient access to downloads, updates, and corrections as they become available. To start the registration process, go to [www.pearsonitcertification.com/register](http://www.pearsonitcertification.com/register) and log in or create an account<sup>\*</sup>. Enter the product ISBN 9780137941582 and click Submit. When the process is complete, you will find any available bonus content under Registered Products.

<sup>\*</sup>Be sure to check the box that you would like to hear from us to receive exclusive discounts on future editions of this product.

# Introduction

There are many reasons to get certified in AWS technology. First of all, AWS certifications validate your AWS cloud knowledge.

To fully understand the AWS cloud, preparing for the AWS Certified Solutions Architect – Associate (SAA-C03) exam is a great place to start. There are other AWS certifications that may be a better fit, depending on your technical level, your current knowledge of cloud concepts, and your current and future jobs with AWS technologies and services. Certifications are broken down into Foundational, Associate, Professional, and Specialty certifications. Full details can be found at

<https://aws.amazon.com/certification/>. AWS frequently adds new certification tracks, but the following are the certifications that are currently available:

- **Foundational:** There is one Foundational certification: AWS Certified Cloud Practitioner. The recommendation is to have at least 6 months of fundamental AWS cloud knowledge before attempting this certification exam. You might be closer to this certification than you think, depending on your current level of technical skills. One advantage of taking the AWS Certified Cloud Practitioner exam first is that it helps you to get used to answering multiple-choice test questions and to learn about the foundational AWS cloud services.

- **Associate:** There are several Associate certifications:
    - **AWS Certified Solutions Architect – Associate:** For individuals working as solutions architects, designing AWS solutions using AWS services
    - **AWS Certified SysOps Administrator – Associate:** For individuals working as systems administrators, managing and operating AWS services
    - **AWS Certified Developer – Associate:** For individuals working as developers, deploying and debugging cloud-based applications hosted at AWS
- Each certification exam expects that you know how the AWS service that you are being tested on works. Each Associate certification has a specific focus:
- **Architect:** The best design possible, based on the question and scenario
  - **SysOps:** The administration steps required to carry out a particular task
  - **Developer:** How to best use the service for the hosted application you are writing
- For example, the three Associate exams would test different aspects of CloudWatch logs:
- **Architect:** The main focus of this exam is on how CloudWatch logs work and the main design features to consider based on specific needs—that is, design

knowledge related to using CloudWatch logs for a variety of solutions.

- **SysOps:** The main focus of this exam is on how to configure CloudWatch logs based on specific needs—that is, configuration and deployment of CloudWatch logs using operational knowledge.
- **Developer:** The main focus of this exam is on what CloudWatch logs are useful for when developing applications for tracking performance of an application hosted on an EC2 instance—that is, knowledge of how a particular AWS service can help in the development and testing process with applications.

Before you attempt one of the Associate certifications, AWS recommends that you have at least 1 year of experience solving problems and implementing solutions using AWS services. AWS really wants to ensure that you have hands-on experience solving problems.

- **Professional:** These certifications include the AWS Certified Solutions Architect Professional and the AWS Certified DevOps Engineer Professional. Professional certifications are not where you normally start your certification journey. AWS recommends that you have at least 2 years of hands-on experience before taking a Professional exam.

- **Specialty:** The Specialty certifications for Advanced Networking, Security, Machine Learning, Data Analytics, SAP on AWS, and Database require advanced knowledge of the subject matter. AWS recommends that you have an Associate certification before you attempt one of these certifications.
- 

### Note

The AWS Certified Solutions Architect – Associate (SAA-C03) certification is globally recognized and does an excellent job of demonstrating that the holder has knowledge and skills across a broad range of AWS topics.

---

### **The Goals of the AWS Certified Solutions Architect – Associate Certification**

The AWS Certified Solutions Architect – Associate certification is intended for individuals who perform in a solutions architect role. This exam validates a candidate's ability to effectively demonstrate knowledge of how to architect and deploy secure and robust applications on AWS technologies. It validates a candidate's ability to

- Have knowledge and skills in the following AWS services: compute, networking, storage, and database and deployment and management services
- Have knowledge and skills in deploying, managing, and operating AWS workloads and implementing security controls and compliance requirements
- Identify which AWS service meets technical requirements
- Define technical requirements for AWS-based applications
- Identify which AWS services meet a given technical requirement

## **Recommended Prerequisite Skills**

While this book provides you with the information required to pass the Certified Solutions Architect – Associate (SAA-C03) exam, Amazon considers ideal candidates to be those who possess the following:

- Experience in AWS technology
- Strong on-premises IT experience
- Understanding of mapping on-premises technology to the cloud
- Experience with other cloud services

## **The Exam Domains**

The AWS Certified Solutions Architect – Associate (SAA-C03) exam is broken down into four major domains. This book covers each of the domains and the task statements.

- **Domain 1: Design Secure Architectures 30%**
  - Task Statement 1: Design secure access to AWS resources
  - Task Statement 2: Design secure workloads and applications
  - Task Statement 3: Determine appropriate data security controls
- **Domain 2: Design Resilient Architectures 26%**
  - Task Statement 1: Design scalable and loosely coupled architectures
  - Task Statement 2: Design highly available and/or fault-tolerant architectures
- **Domain 3: Design High-Performing Architectures 24%**
  - Task Statement 1: Determine high-performing and/or scalable storage solutions
  - Task Statement 2: Design high-performing and elastic compute solutions
  - Task Statement 3: Determine high-performing database solutions
  - Task Statement 4: Determine high-performing and/or scalable network architectures

- Task Statement 5: Determine high-performing data ingestion and transformation solutions
- **Domain 4: Design Cost-Optimized Architectures 20%**
  - Task Statement 1: Design cost-optimized storage solutions
  - Task Statement 2: Design cost-optimized compute solutions
  - Task Statement 3: Design cost-optimized database solutions
  - Task Statement 4: Design cost-optimized network architectures

## **Steps to Becoming an AWS Certified Solutions Architect – Associate**

To become an AWS Certified Solutions Architect – Associate, an exam candidate must meet certain prerequisites and follow specific procedures. Exam candidates must ensure that they have the necessary background and technical experience for the exam and then sign up for the exam.

### **Signing Up for the Exam**

The steps required to sign up for the AWS Certified Solutions Architect – Associate exam are as follows:

**Step 1.** Create an AWS Certification account at  
<https://www.aws.training/Certification> and schedule your exam

from the home page by clicking Schedule New Exam.

**Step 2.** Select a testing provider, either Pearson VUE or PSI, and select whether you want to take the exam at a local testing center or online from your home or office. If you choose to take an online exam, you will have to agree to the online testing policies.

**Step 3.** Complete the examination signup by selecting the preferred language and the date of your exam.

**Step 4.** Submit the examination fee.

---

Tip

Refer to the AWS Certification site at  
<https://aws.amazon.com/certification/>for more information regarding this and other AWS certifications.

---

## How to Use This Book

This book maps directly to the domains of the AWS Certified Solutions Architect – Associate (SAA-C03) exam and includes a number of features that help you understand the topics and prepare for the exam.

## **Objectives and Methods**

This book uses several key methodologies to help you discover the exam topics on which you need more review, to help you fully understand and remember those details, and to help you ensure that you have retained your knowledge of those topics. This book does not try to help you pass the exam only by memorization; it seeks to help you truly learn and understand the topics. This book is designed to help you pass the AWS Certified Solutions Architect – Associate (SAA-C03) exam by using the following methods:

- Helping you discover which exam topics you have not mastered
- Providing explanations and information to fill in your knowledge gaps
- Supplying exercises that enhance your ability to recall and deduce the answers to test questions
- Providing practice exercises on the topics and the testing process via test questions on the companion website

## **Book Features**

To help you customize your study time using this book, the core chapters have several features that help you make the best use of your time:

- **Foundation Topics:** The sections under “Foundation Topics” describe the core topics of each chapter.
- **Exam Preparation Tasks:** The “Exam Preparation Tasks” section lists a series of study activities that you should do at the end of each chapter:
  - **Review All Key Topics:** The Key Topic icon appears next to the most important items in the “Foundation Topics” section of the chapter. The “Review All Key Topics” activity lists the key topics from the chapter, along with the number of the page where you can find more information about each one. Although the contents of the entire chapter could be tested on the exam, you should definitely know the information listed in each key topic, so you should review these.
  - **Define Key Terms:** Although the AWS Certified Solutions Architect – Associate (SAA-C03) exam may be unlikely to word a question “Define this term,” the exam does require that you learn and know a lot of terminology. This section lists the most important terms from the chapter and asks you to write a short definition and compare your answer to the glossary at the end of the book.
  - **Q&A:** Confirm that you understand the content that you just covered by answering these questions and reading the answer explanations.

- **Web-based practice exam:** The companion website includes the Pearson Test Prep practice test engine, which enables you to take practice exam questions. Use it to prepare with a sample exam and to pinpoint topics where you need more study.

## How This Book Is Organized

This book contains 14 core chapters—[Chapters 2](#) through [15](#). [Chapter 1](#) introduces the foundations of AWS, and [Chapter 16](#) provides preparation tips and suggestions for how to approach the exam. Each core chapter covers a specific task statement or multiple task statements of the domains for the AWS Certified Solutions Architect – Associate (SAA-C03) exam.

## Companion Website

Register this book to get access to the Pearson Test Prep practice test software and other study materials plus additional bonus content. Check this site regularly for new and updated postings written by the author that provide further insight into the more troublesome topics on the exam. Be sure to check the box indicating that you would like to hear from us to receive updates and exclusive discounts on future editions of this product or related products.

To access this companion website, follow these steps:

**Step 1.** Go to <https://www.pearsonitcertification.com/register> and log in or create a new account.

**Step 2.** Enter the ISBN 9780137941582.

**Step 3.** Answer the challenge question as proof of purchase.

**Step 4.** Click the Access Bonus Content link in the Registered Products section of your account page to be taken to the page where your downloadable content is available.

Please note that many of our companion content files can be very large, especially image and video files.

If you are unable to locate the files for this title by following these steps, please visit

<https://www.pearsonITcertification.com/contact> and select the Site Problems/Comments option from the Select a Topic drop-down list. Our customer service representatives will assist you.

## Pearson Test Prep Practice Test Software

As noted earlier, the Pearson Test Prep practice test software comes with two full practice exams. These practice exams are available to you either online or as an offline Windows application. To access the practice exams that were developed with this book, see the instructions in the card inserted in the sleeve at the back of the book. This card includes a unique access code that enables you to activate your exams in the Pearson Test Prep practice test software. For more information about the practice exams and more tools for exam preparation, see [Chapter 16](#).

# Figure Credits

Cover: Yurchanka Siarhei/Shutterstock

Chapter opener: Charlie Edwards/Getty Images

Figures 1.1, 1.3 through 1.6, 1.10, 1.12 through 1.4, 2.1 through 2.4, 2.6 through 2.8, 2.13, 2.14, 3.1 through 3.4, 3.7 through 3.9, 3.11 through 3.24, 3.27 through 3.37, 3.39 through 3.48, 4.3, 4.4, 4.6 through 4.8, 4.11 through 4.14, 4.22 through 4.34, 5.2, 5.6 through 5.11, 5.14 through 5.16, 5.18, 6.7, 6.11 through 6.15, 6.17 through 6.20, 6.22, 6.23, 6.26 through 6.30, 7.5, 7.11 through 7.14, 7.33, 7.34, 8.1 through 8.13, 8.15, 8.17 through 8.23, 9.2 through 9.5, 9.7, 9.9, 9.10, 9.12, 9.13 through 9.29, 10.1, 10.4, 10.10 through 10.12, 10.17, 10.18, 11.3 through 11.7, 11.10 through 11.21, 11.23, 11.24, 11.27 through 11.31, 11.33, 11.34, 12.1 through 12.10, 12.12 through 12.17, 13.3 through 13.12, 14.1, 14.3 through 14.6, 14.13, 15.4, 16.1, 16.2: Amazon Web Services, Inc

Figure 2.11: Adam Wiggins

Figures 2.9a, 7.1: Andrei Minsk/Shutterstock

Figures 3.10, 3.38, 11.25: Microsoft Corporation

# Chapter 1

## Understanding the Foundations of AWS Architecture

This chapter covers the following topics:

- [Essential Characteristics of AWS Cloud Computing](#)
- [AWS Cloud Computing and NIST](#)
- [Moving to AWS](#)
- [Operational Benefits of AWS](#)
- [Cloud Provider Responsibilities](#)
- [Security at AWS](#)
- [Migrating Applications](#)
- [The AWS Well-Architected Framework](#)
- [AWS Services Cheat Sheet](#)

The AWS Certified Solutions Architect – Associate (SAA-C03) exam that we are discussing in this book measures your technical competence in architecting workloads to run successfully in the Amazon Web Services (AWS) cloud. For any of their associate certification exams, AWS does not expect you to be an expert in every single cloud service, as that is an impossible task. However, AWS does expect you to be able to display a high level of competence about how to architect

(design, deploy, monitor, and manage) workloads running on AWS cloud architecture based on the exam domains of knowledge. You can find the SAA-C03 exam guide here: [https://d1.awsstatic.com/training-and-certification/docs-sa-assoc/AWS-Certified-Solutions-Architect-Associate\\_Exam-Guide\\_C03.pdf](https://d1.awsstatic.com/training-and-certification/docs-sa-assoc/AWS-Certified-Solutions-Architect-Associate_Exam-Guide_C03.pdf). The SAA-C03 exam guide lists the AWS services that could be tested on the exam, and what AWS services are not covered.

The goal of writing this book is to include enough technical details for all readers to absorb and pass the AWS Certified Solutions Architect – Associate (SAA-C03) exam. The following list should help you to gauge whether you should read this entire chapter or skim through the topics:

- If you are coming from a technical background but don't know anything about the AWS cloud, start with this first chapter and read it carefully.
- If you have a background working in the AWS cloud but this is your first certification attempt, you might not need to read the entire chapter, but you should review the first chapter's content, and study the final section, "AWS Services Cheat Sheet."
- If you already are certified as an AWS Certified Solutions Architect – Associate and it's time to re-certify, you might not

need to read this chapter, but you should study the final section, “AWS Services Cheat Sheet,” to ensure that you’re up to speed on the latest AWS services covered on the exam.

And let’s be clear, the goal of this book is to help you pass the AWS Certified Solutions Architect – Associate exam. If you ace the exam, great! However, passing the exam should be your overall goal. You need to get roughly 72% of the exam questions right to pass the exam; Amazon is not clear as to the exact percentage for passing the exam but it’s in this range. The AWS SAA-C03 exam is 65 multiple choice questions. However, it’s very important to understand that 15 of the 65 exam questions are beta questions that don’t count! Therefore, there are 50 questions you must answer successfully. Answering approximately 37 questions correctly out of the 50 questions that count will achieve your goal of becoming an AWS Certified Solutions Architect – Associate.

The SAA-C03 exam is marked using what is defined as *scaled scoring*. The questions that you are presented on your exam most likely will not be the same as those presented to other exam candidates; the difficulty of each exam question is weighted to ensure the total knowledge level of each exam as a whole is maintained. Additional details on how to prepare to

take the exam are fully covered in the last chapter of this book, [Chapter 16, “Final Preparation.”](#)

The following list of tasks will also help you greatly in the goal of becoming certified:

- **Read the FAQs:** Each AWS cloud service has a frequently asked questions (FAQs) summary that summarizes the service and its highlights. When learning about an AWS service, always start with the FAQ—you won’t be disappointed. And be sure to take notes as you learn.
- **Read the AWS Well-Architected Framework PDFs:** The exam is based on the AWS Well-Architected Framework. Reading the PDF of each pillar is a great study aid for understanding the mindset of the exam questions, and will also prepare you to be a great AWS consultant/cloud architect. Make sure to review the Security Pillar, Reliability Pillar, Performance Efficiency Pillar, and the Cost Optimization Pillar. See <https://aws.amazon.com/architecture/well-architected/>.
- **Sign up for a free AWS cloud account:** This is the best method to practice hands-on tasks for the exam. Create multiple AWS accounts; you are not limited to one free AWS account, but a different e-mail address must be used as the root login for each AWS account that is created.

- **Complete AWS Well-Architected Labs:** Complete as many of the labs as possible that relate to the AWS Certified Solutions Architect – Associate exam topics. The labs are foundational (100), intermediate (200), and advanced (300), as partially shown in [Figure 1-1](#) for the Security category. See <https://wellarchitectedlabs.com/>.

The figure shows a screenshot of the AWS Well-Architected Framework Hands-on Labs interface. At the top, there are three main categories represented by icons: 'Identity & access management' (a user icon), 'Detection' (a magnifying glass icon), and 'Infrastructure protection' (a shield icon with a checkmark). Below these categories, the title 'Labs & Quests' is displayed in large, bold, black font. To the right of the title is a detailed list of labs categorized into two levels: '100 Level Foundational Labs' and '200 Level Intermediate Labs'. The '100 Level Foundational Labs' section includes the following items:

- AWS Account Setup and Root User
- Creating your first Identity and Access Management User, Group, Role
- CloudFront with S3 Bucket Origin
- Enable Security Hub
- Create a Data Bunker Account

The '200 Level Intermediate Labs' section includes the following items:

- Automated Deployment of Detective Controls
- Automated Deployment of EC2 Web Application
- Automated Deployment of IAM Groups and Roles
- Level 200: Automated Deployment of VPC
- Level 200: Automated Deployment of Web Application Firewall
- Level 200: Automated IAM User Cleanup
- Level 200: Basic EC2 Web Application Firewall Protection

- 100 Level Foundational Labs
  - AWS Account Setup and Root User
  - Creating your first Identity and Access Management User, Group, Role
  - CloudFront with S3 Bucket Origin
  - Enable Security Hub
  - Create a Data Bunker Account
- 200 Level Intermediate Labs
  - Automated Deployment of Detective Controls
  - Automated Deployment of EC2 Web Application
  - Automated Deployment of IAM Groups and Roles
  - Level 200: Automated Deployment of VPC
  - Level 200: Automated Deployment of Web Application Firewall
  - Level 200: Automated IAM User Cleanup
  - Level 200: Basic EC2 Web Application Firewall Protection

**Figure 1-1** AWS Well-Architected Framework Hands-on Labs

- **Use the AWS Well-Architected Tool:** The AWS Well-Architected Tool is a self-paced utility that consists of Well-Architected Framework questions from each pillar to make you consider which best practices and procedures should be considered when hosting your workloads at AWS. This is a great study aid for the exam, available at  
<https://www.wellarchitectedlabs.com/well-architectedtool/>.
- **Complete the AWS security workshops:** AWS offers a variety of security workshops that will help you understand AWS security best practices; see  
<https://awssecworkshops.com/>.
- **Answer as many sample exam questions as you can:**  
Included in this book is a test engine with hundreds of test questions. The hardest part of preparing to take the exam is getting used to answering multiple-choice test questions. The more practice you have, the better you will be prepared. AWS also has some sample questions for the SAA-C03 exam here:  
[https://d1.awsstatic.com/training-and-certification/docs-sa-assoc/AWS-Certified-Solutions-Architect-Associate\\_Sample-Questions.pdf](https://d1.awsstatic.com/training-and-certification/docs-sa-assoc/AWS-Certified-Solutions-Architect-Associate_Sample-Questions.pdf)  
and here:  
<https://explore.skillbuilder.aws/learn/course/external/view/earning/13266/aws-certified-solutions-architect-associate->

[official-practice-question-set-saa-c03-english?](#)

[saa=sec&sec=prep](#)

- **Browse the AWS Architecture Center:** The AWS Architecture Center (<https://aws.amazon.com/architecture/>) has many examples of how to deploy reference architecture for analytics, compute and HPC deployments, and databases, to name just a few. Walking through the step-by-step notes provides a great overview of the associated AWS services and can be helpful in visualizing how AWS architecture is designed and deployed.

## Essential Characteristics of AWS Cloud Computing

In 2021, CEO Andy Jassy estimated that the cloud was currently less than 5% of global IT spending, which suggests that moving workloads to the cloud for many companies is really just beginning. The public cloud providers AWS and Microsoft Azure have been established for well over a decade and have strong infrastructure as a service (IaaS) and platform as a service (PaaS) offerings available around the world. Google Cloud Platform (GCP), Oracle Cloud, and IBM Cloud are also viable alternatives. [Figure 1-2](#) shows the Gartner Magic Quadrant for Cloud Infrastructure and Platform Services (see <https://www.gartner.com/en/research/methodologies/magic-quadrants-research>), which indicates the current favorite cloud

technology providers companies can choose to align with. In the Leaders quadrant, Amazon Web Services leads, followed closely by Microsoft and then Google. Alibaba Cloud aligns with the Visionaries quadrant, and Oracle, Tencent Cloud, and IBM currently occupy the Niche Players quadrant.

When I started my career as a computer technician in the 1990s, most corporations that I supported used several computer-based services running on mainframes that were not located on premises. Accounting services were accessed through a fast (at the time) 1200-baud modem that was connected using one of those green-screened digital terminals. The serial cable, threaded through the drop ceiling to connect the terminal, was strong enough to pull a car.

Today we rely more and more on one or more public cloud providers for hosting many types of workloads on an ever-increasing collection of very specialized data centers and cloud services. There is no hardware ownership, the cloud provider owns the services, and customers rent cloud services as required.



Source: Gartner (July 2021)

**Figure 1-2** Gartner's Magic Quadrant of Top Public Cloud Providers  
(<https://www.gartner.com/en/research/methodologies/magic-quadrants-research>)<sup>1</sup>

<sup>1</sup> Gartner does not endorse any vendor, product, or service depicted in its research publications and does not advise technology users to select only those vendors with the highest ratings or other designation. Gartner research publications consist of the opinions of Gartner's research organization and should not be construed as statements of fact. Gartner disclaims all warranties, expressed or implied, with respect to this research, including any warranties of merchantability or fitness for a particular purpose.

You might think that the public cloud only offers virtual resources, but the AWS cloud and others *can* provide bare-metal servers if requested. AWS will happily host your applications and databases on bare-metal servers hosted at AWS, or in your own data centers. Of course, more commonly, AWS offers you a wide variety of virtual servers in many different sizes and designs. AWS is also quite happy if you continue to operate your on-premises data centers and coexist with cloud resources and services operating at AWS. AWS also offers AWS Outposts, which enables customers to run an ever-increasing number of AWS cloud services on premises. Microsoft Azure will offer to sell you a copy of its complete

Azure cloud operating system, called Azure Stack, installed on servers in your data centers. It's getting harder to define the public cloud these days.

Applications that are hosted in the public cloud leverage virtual server, network, and storage resources combined with cloud services that provide monitoring, backup services, and more. Hardware devices, such as routers, switches, and storage arrays, have been replaced by AWS-managed cloud services built from the same virtual computers, storage, and networking components used by AWS themselves that are offered to each customer. This doesn't mean that companies aren't still using hardware devices on premises. However, it is possible to run hundreds or thousands of virtual machines in parallel, outperforming the functionality of a single hardware switch or router device. Most AWS cloud services are hosted on virtual machines called Amazon Elastic Cloud Compute (EC2) instances running in massive server farms powering the storage arrays, networking services, load-balancing, and auto-scaling services provided by AWS are part of Amazon Web Services (AWS). For example, AWS Config helps you manage compliance, and the AWS Backup service backs up AWS storage services.

## **AWS Cloud Computing and NIST**

If you haven't heard of the National Institute of Standards and Technology (NIST), a branch of the U.S. government, you're not alone. Around 2010, NIST began documenting the emerging public cloud. After consulting the major cloud vendors, it released an initial report in June 2011, Special Publication 800-145, "The NIST Definition of Cloud Computing," defining the cloud services that were common across all public cloud vendors. The report's genius is that it defined in 2011 what the emerging public cloud actually became. NIST's cloud definitions have moved from mere definitions, to accepted standards that are followed by all of the public clouds we use today.

The five key NIST definitions of the public cloud have morphed into a definitive standard methodology of how cloud providers and thousands of customers operate in the public cloud. The report can be found here:

<https://nvlpubs.nist.gov/nistpubs/legacy/sp/nistspecialpublication800-145.pdf>. The five essential characteristics of the cloud model defined by NIST are

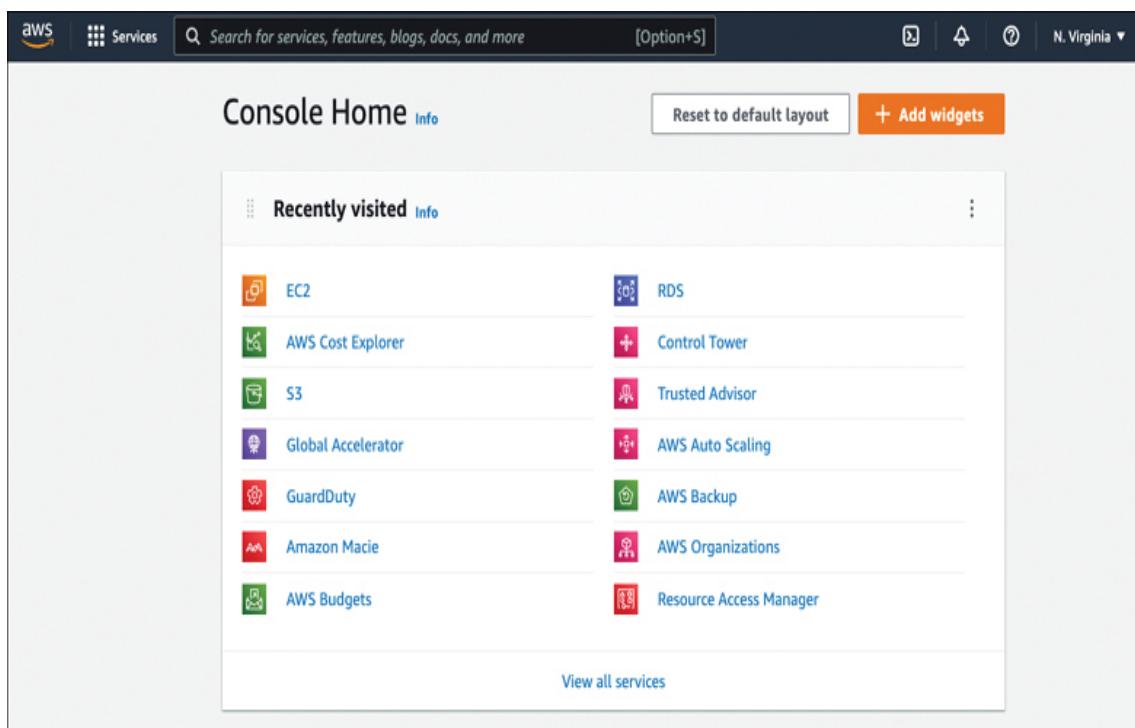
- On-demand Self-Service
- Broad Network Access
- Resource Pooling
- Rapid Elasticity
- Measured Service

The sections that follow describe these essential NIST characteristics.

## On-Demand Self-Service

These days companies don't just *expect* cloud service to be delivered quickly; they *demand* it.

Every cloud provider, including AWS, offers a self-service management portal (see [Figure 1-3](#)). Request any cloud service, and in seconds, or minutes, it's available in your AWS account, ready to be configured or used. Gone are the days of requesting a virtual server via email and waiting several days until it's available. At AWS, a virtual server can be ordered and operational in under 5 minutes. Creating and using an Amazon Simple Storage Service (Amazon S3) bucket is possible within seconds. It is also possible to procure a software-defined network (called an Amazon Virtual Private Cloud) and have it operational in seconds. Using the AWS management console enables customers to order and configure many cloud services across many AWS regions. Any cloud service ordered is quickly delivered using automated procedures running in the background.



**Figure 1-3** The AWS Management Console

## Broad Network Access

Cloud services running at AWS can be accessed from anywhere there is an Internet connection, using just a web browser. AWS provides secure HTTPS endpoints to access every cloud service hosted at AWS. However, your company might not want or require what NIST defined as broad network access, which is public Internet network access to your workloads. Many companies that are moving to the AWS cloud have no interest in a publicly accessible software solution. They want their hosted cloud services to remain private, accessible only by their

employees using private network connections. Each cloud customer ultimately defines their definition of broad network access: public Internet connections, private VPN or fiber connections, or both.

At AWS, applications and services can be made publicly available, or they can remain completely private. Virtual private network (VPN) connections from your place of work to AWS are commonplace. Customers can also order an AWS Direct Connect connection, a private fiber connection to AWS resources running at speeds up to 100 Gbps. Depending on the type of application you're hosting in the AWS cloud, high-speed network access may be essential.

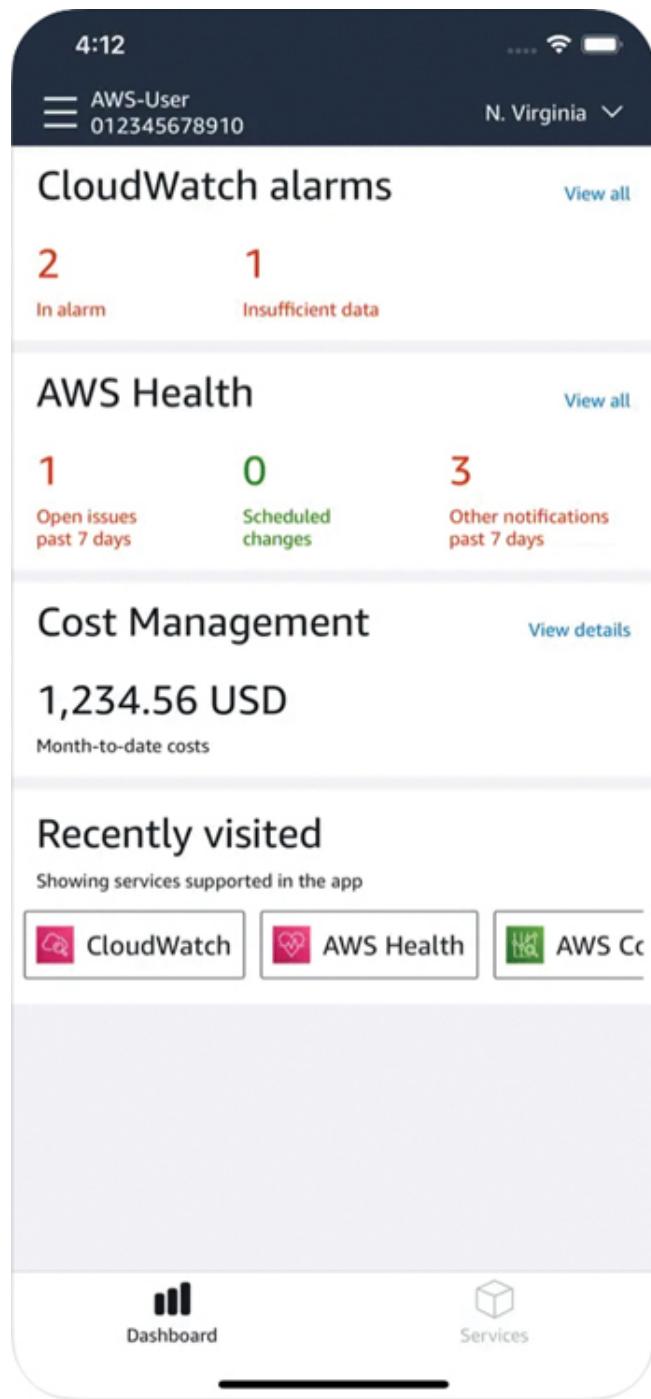
It's also possible to administer AWS services from a smartphone by using an AWS app (see [Figure 1-4](#)). Certainly, accessing AWS from any device is possible.

## Resource Pooling

Infrastructure resources for AWS cloud services are located across different geographical regions of the world in many data centers. A company running an on-premises private cloud will typically pool its virtual machines, memory, processing, and

networking capabilities into one or two data centers offering a limited pool of compute and network resources.

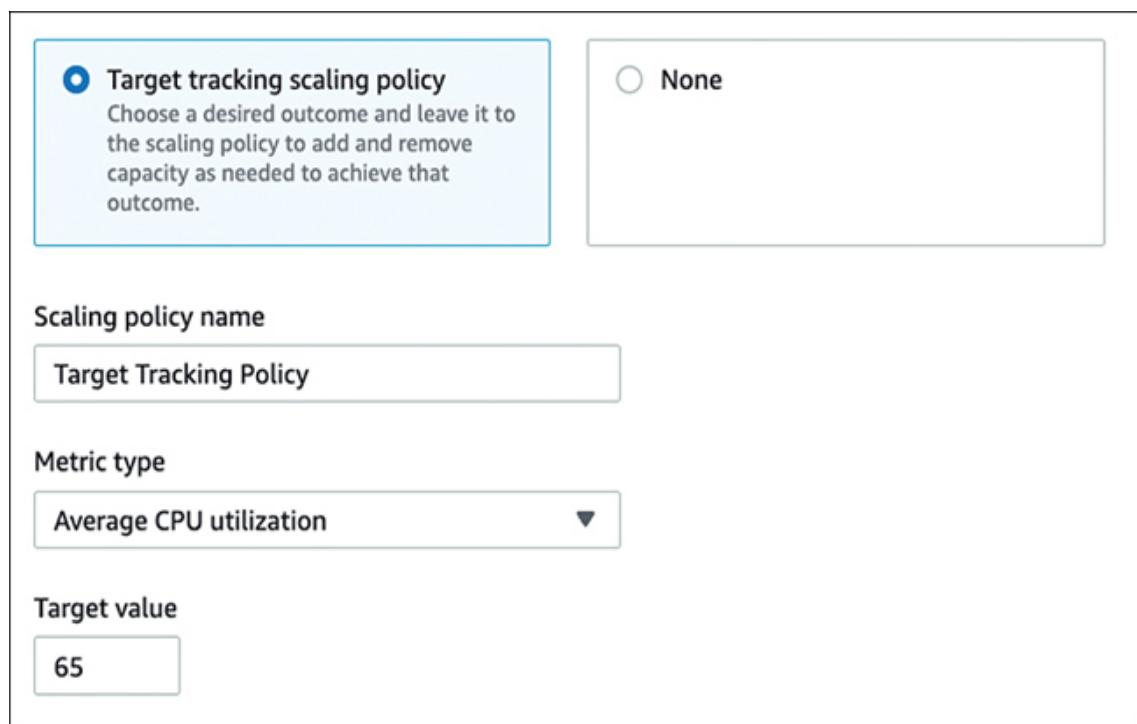
AWS has clusters of data centers, stored in multiple availability zones (AZs) across each region, and each AZ has thousands of bare-metal servers and storage resources available and online, allowing customers to host their workloads with a high level of resiliency and availability. Without a massive pool of compute resources, AWS would not be able to allow customers to dynamically allocate compute resources to match their performance requirements and workload needs. Amazon S3 object storage is offered as unlimited; there is no defined maximum storage limit.



**Figure 1-4** AWS Apps for Phone

## Rapid Elasticity

Rapid elasticity in the public cloud is *the* key feature for hosted cloud applications. At AWS, compute and storage resources are defined as elastic. Workloads running in the AWS cloud for Amazon EC2 instances or Amazon Elastic Container Service (Amazon ECS) deployments have the capability to automatically scale using a scaling policy to dynamically resize an Auto Scaling group of web or application servers using several scaling policies, including target tracking (see [Figure 1-5](#)). In this example, EC2 Auto Scale will maintain CPU utilization of 65%; additional compute resources will be automatically added or removed to maintain the desired target value.



**Figure 1-5** Workload Scaling Based on Demand

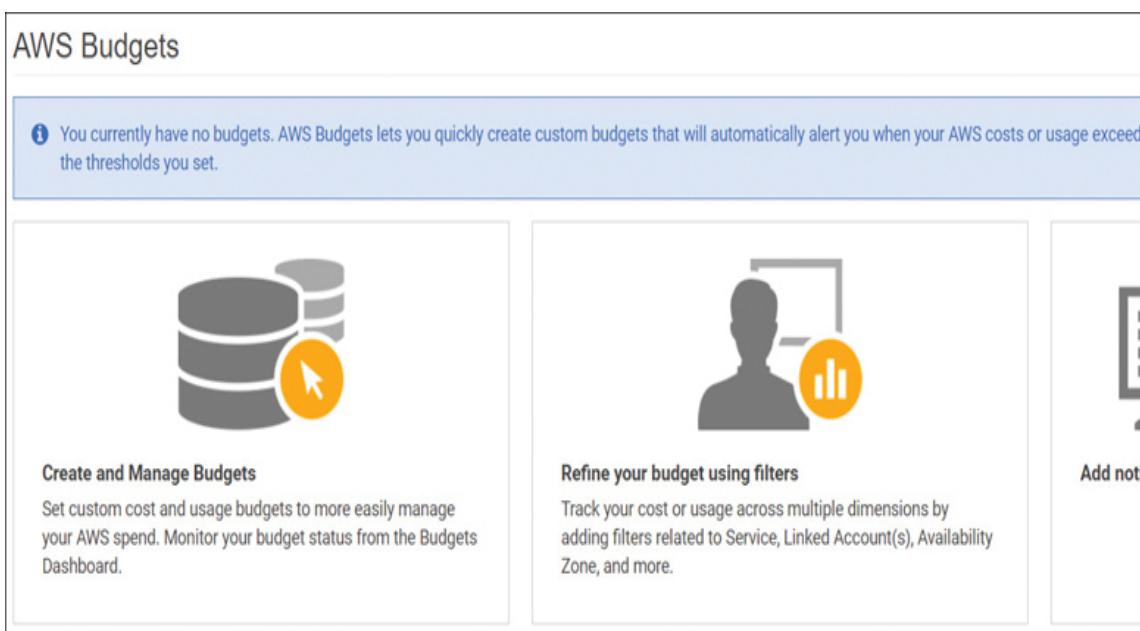
Elasticity—that is, dynamic scaling—is an automated solution scaling compute resources up or down in size based on workload needs. Administrators these days don’t need to turn off virtual servers, add additional RAM, and turn the servers back on again; instead, they can deploy *horizontal scaling*—that is, automatically add or remove additional servers as required. AWS EC2 Auto Scaling is integrated with the Amazon CloudWatch monitoring service using metrics and event-driven alarms to dynamically increase or decrease compute resources as required.

## Measured Service

In the AWS cloud, you are billed for only the services that you use or consume; this concept is referred to as a *measured service*. AWS charges can be broken down into compute, storage, and data transfer charges. Packet flowing inbound (i.e., ingress to the AWS cloud) is usually free. By contrast, outbound packet flow (i.e., egress traffic across the Internet, a private network connection, or network replication traffic between a primary and alternate database server hosted on subnets in different availability zones) is charged an outbound data transfer fee. In the case of computer services such as AWS EC2 compute instances, charges are per hour for EC2 usage calculated by the second based on the size of the EC2 instance,

operating system, and the AWS Region where the instance is launched. For storage services such as Amazon S3 storage or virtual hard drives (Amazon EBS), storage charges are per gigabyte used per month.

If a cloud service in your AWS account is on, charges will apply. Running hosted workloads in the AWS cloud requires a detailed understanding of how costs are charged; the management of costs at AWS is one of the most important tasks to understand and control. AWS has many useful tools to help you control your cloud costs, including the AWS Simple Monthly Calculator, AWS Cost Explorer, and AWS Budgets (see [Figure 1-6](#)).



**Figure 1-6** Using AWS Budgets to Track and Alert When Costs Are Over Budget

Being billed for consuming cloud services is a reality that we are all personally used to; for example, Netflix, Disney, and Dropbox are common services. However, billing at AWS is different from the flat per-month fees for personal software as a service (SaaS) services. Customers must understand and carefully monitor their compute, storage, and data transfer costs or else their monthly charges can become extremely expensive. For example, a load balancer can be ordered at AWS for approximately \$18 per month. However, the data traffic transferred through the load balancer is also charged, so the overall monthly price could be substantial.

## Moving to AWS

Once an organization has decided to move to the AWS cloud, countless moving parts begin to churn. People need to be trained, infrastructure changes must take place, developers need to develop applications with a different mindset, and administrators must get up to speed. Generally, people at companies beginning to utilize cloud services typically have several mindsets:

- **The corporate mentality:** You currently have data centers, infrastructure, and virtualized applications. Ever-increasing infrastructure and maintenance costs are driving you to look

at what options are available in the AWS cloud. Your starting point could be to utilize the available IaaS offerings for servers, storage, monitoring, and networking services.

- **The born-in-the-cloud mentality:** You're a developer (or a nimble organization) with a great idea but not much startup funding. You also don't have a local data center, and want to get going as soon as possible. Your starting point could be to utilize the available IaaS offerings for servers, storage, monitoring, and networking, and the PaaS offerings, to speed up the development process.
- **The startup mentality:** You've just lost your job due to a merger or buyout and are determined to strike out on your own. Your brand-new company has no data center and lacks cash, but it has plenty of ideas. Your starting point will be the same as the born-in-the-cloud mentality example.

Each of these starting mindsets or outlooks will have differing points of view about how to migrate or design their cloud infrastructure and hosted applications. If you come from a corporate environment, you will probably expect the cloud provider to have a detailed service-level agreement (SLA) that you can change to match your needs. You will also probably have expectations about how much detail should be provided about the cloud provider's infrastructure and cloud services.

AWS has service-level agreements for its cloud services and very detailed documentation for each hosted cloud service.

---

### Note

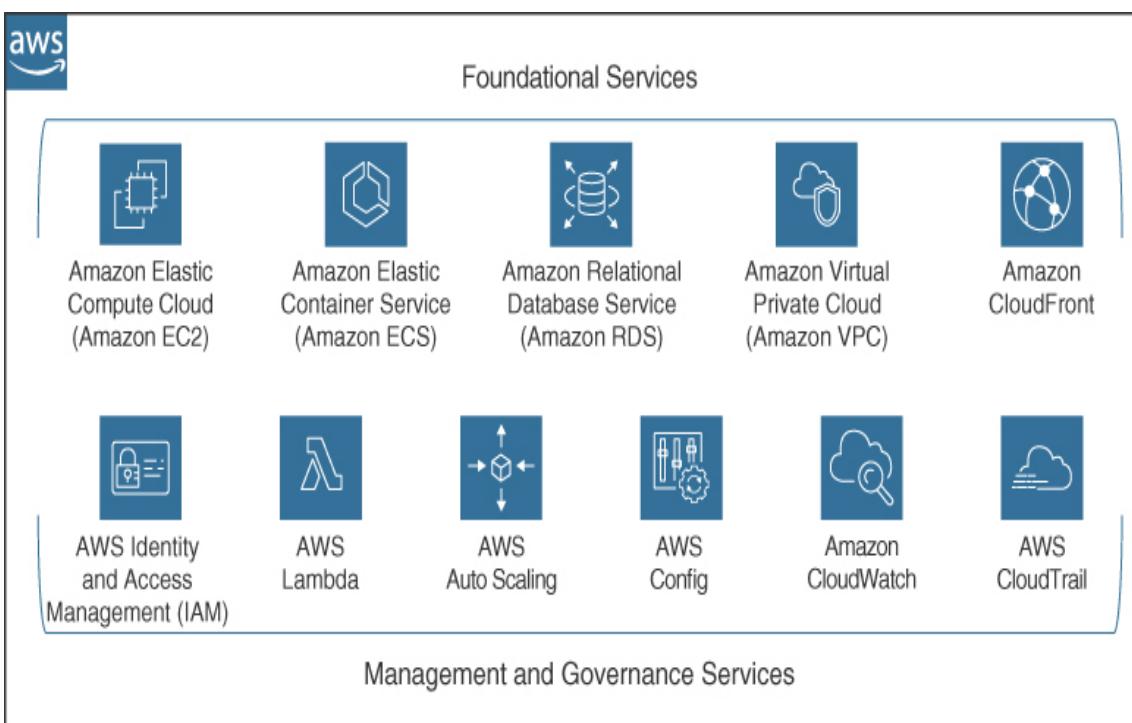
AWS has options for developers who want to craft and deploy applications hosted at AWS. Visit <https://aws.amazon.com/startups/> for further information about how you might be able to qualify for AWS Promotional Credit. There's a possibility of getting up to \$15,000 in credits over 2 years, including AWS support and training.

---

## Infrastructure as a Service (IaaS)

Many cloud services offered by AWS are defined as IaaS services, and are defined in this book as foundational services that are used by every customer (see [Figure 1-7](#)). Virtualized servers (Amazon EC2), container services (Amazon ECS), and database services (Amazon RDS) are hosted on a fast private software-defined network (SDN). Each customer's IaaS services are isolated from all other AWS customers by default. A robust security service named AWS Identity and Access Management (IAM) enables each customer to secure and control every ordered IaaS service as desired. A wide variety of supporting

services, defined as Management and Governance services, also shown in [Figure 1-7](#), provide monitoring (Amazon CloudWatch), audit services (AWS CloudTrail), scaling of compute resources (AWS Auto Scaling), governance (AWS Config), and event-driven automation (AWS Lambda).



**Figure 1-7** Infrastructure as a Service at AWS

Hosting compute workloads at AWS requires the creation of a network environment called Amazon Virtual Private Cloud (VPC) hosting web, application, and database services on subnets. Customers have the flexibility to create whatever architectural stack is required at AWS, using the vast number of

IaaS services and management services available. Many companies moving to AWS typically start with IaaS services, because the IaaS services at AWS closely mirror their current on-premises virtual environment.

Here are some examples of the essential cloud services at AWS:

- **Compute services:** The previously introduced Amazon EC2 is a cloud service that provides virtual servers (dedicated, multi-tenant, or bare-metal) in an ever-increasing variety of options. Amazon Elastic Container Service (Amazon ECS) supports Docker containers running at AWS, or on-premises using AWS Outpost deployments. Amazon Elastic Kubernetes Service (EKS) supports Kubernetes deployments at AWS or on-premises using AWS Outposts.
- **Storage services:** Amazon S3 is a cloud service that provides unlimited object storage in Amazon S3 buckets or archived storage in vaults. There are shared storage arrays: Amazon Elastic File System (Amazon EFS) for Linux, and Amazon FSx for Windows File Server for Microsoft Windows deployments, and virtual block storage volumes using the Amazon Elastic Block Store (Amazon EBS) service.
- **Database services:** AWS offers a fully managed database service called Amazon Relational Database Service (Amazon RDS). Choose from Amazon Aurora (with MySQL or

PostgreSQL compatibility), MySQL, PostgreSQL, Oracle, and Microsoft SQL Server engines. Using Amazon RDS, AWS builds, hosts, maintains, backs up, and synchronizes HA pairs or clusters of primary/standby database servers, leaving customers the single task of managing their data records.

Many other managed database services are also available at AWS, including Amazon DynamoDB, a NoSQL database; and Amazon ElastiCache, a managed in-memory caching service that supports Memcached and Redis deployments.

- **Automating AWS infrastructure:** AWS CloudFormation enables customers to automate the process of modeling and provisioning infrastructure stacks, complete with the required compute, storage, networks, load balancers, and third-party resources required for each workload. Template files are created using either JSON or YAML declarative code.
- **Auditing:** AWS CloudTrail is enabled in every AWS account, tracking and recording all application programming interface (API) calls and authentication calls. Customers can also configure AWS CloudTrail to store audit information in Amazon S3 Glacier archive forever.
- **Monitoring:** AWS CloudWatch is a powerful monitoring service with metrics for more than 70 AWS services that can be used to monitor resources and application operations

using alarms to carry out automated actions when predetermined thresholds are breached.

- **VMware Cloud on AWS:** Many companies use VMware ESXi infrastructure for their on-premises application servers. Capital expenses and licensing costs are some of the biggest expenses incurred when running an ever-expanding on-premises private cloud. Virtualization was supposed to be the answer to controlling a company's infrastructure costs; however, the cost of hosting, running, and maintaining virtualization services became extremely high as deployments expand in size and complexity. Replacing on-premises VMware deployments with AWS-hosted virtualized servers running on AWS's hypervisor services removes a company's need for hypervisor administration expertise. Many applications used by corporations are also now widely available in the public cloud as hosted applications defined as a software as a service (SaaS) application. VMware ESXi is also available as VMware Cloud on AWS, using VMware's software-defined data center architecture running on AWS infrastructure.

---

### Note

At AWS, infrastructure and platform services and resources are spread across the world in 31

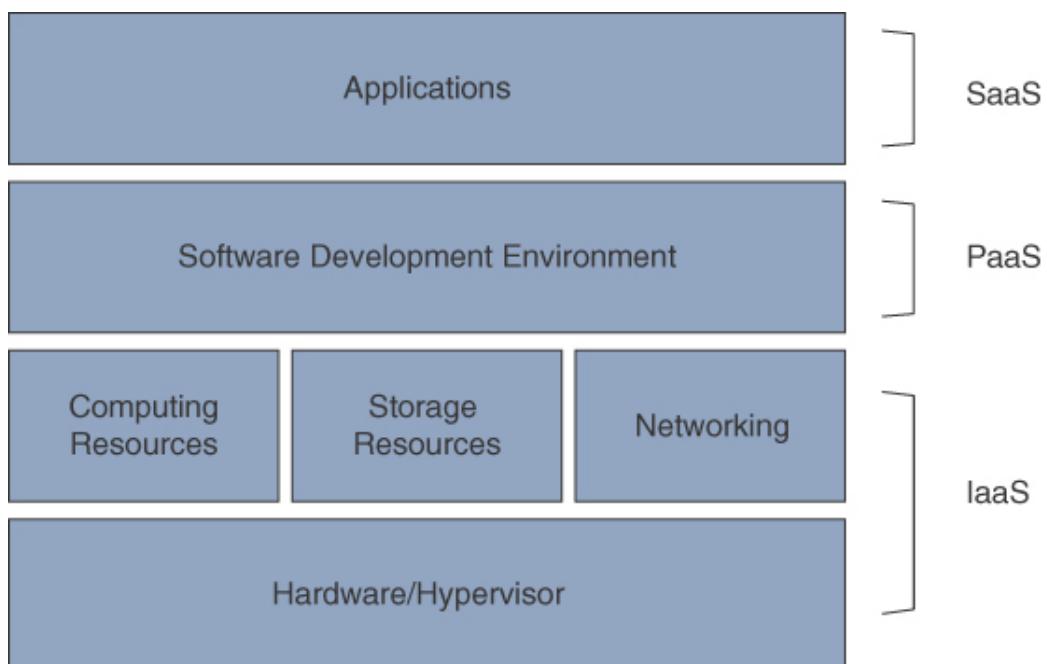
different regions (2022), and additional regions are scheduled to be added. If you are in a large population center, the odds are that access to AWS cloud resources is close by. If AWS is not yet close by, you still might be able to connect using an edge location or a local point of presence connection. To review the current AWS infrastructure, visit [https://aws.amazon.com/about-aws/global-infrastructure/regions\\_az/](https://aws.amazon.com/about-aws/global-infrastructure/regions_az/).

---

## Platform as a Service (PaaS)

PaaS cloud providers enable your company's developers to create custom applications on a variety of popular development platforms, such as Java, PHP, and Python and Go. Your choice of language and development framework will determine the PaaS vendor you select. Using a PaaS provider means that developers don't have to manually build and manage the infrastructure components required for each workload; instead, the required infrastructure resources for each workload running in the development, testing, and production environments are created, hosted, and managed by the PaaS cloud provider. After an application has been developed and tested and is ready for production, end users can access the application using the

application's public URL. In the background, the PaaS cloud provider hosts and scales the hosted SaaS workload based on demand. As the number of users using the workload changes, the infrastructure resources scale out or in as required. PaaS environments are installed on the IaaS resources of the PaaS cloud provider, as shown in [Figure 1-8](#). In fact, IaaS is always behind all “as a service” monikers. Examples of PaaS providers include Google Cloud, Cloud Foundry, and Heroku.



**Figure 1-8** IaaS Hosting the PaaS Layer

The Cloud Foundry PaaS solution is offered for application development at IBM Cloud, running a customized version of the Cloud Foundry platform components. Developers can sign up

and focus on writing applications. All application requests are handled by the PaaS layer interfacing with the IaaS layer, where the application's compute, storage, load-balancing, and scaling services operate.

Another popular solution for developing applications in the public cloud, Heroku, a container-based PaaS environment, enables developers to create and run applications using a variety of development platforms. Just as with IBM Cloud, once the workload is deployed into production, Heroku hosts, load-balances, and auto-scales each workload as required and sends each customer a bill for the infrastructure hosting costs used each month.

When developing applications at a PaaS provider, remember that programming languages change from time to time; therefore, the associated APIs offered by each cloud provider can change as well—and sometimes without much warning. Developers and companies must keep up to date with any ongoing changes or there can be issues when using a cloud-hosted PaaS development platform.

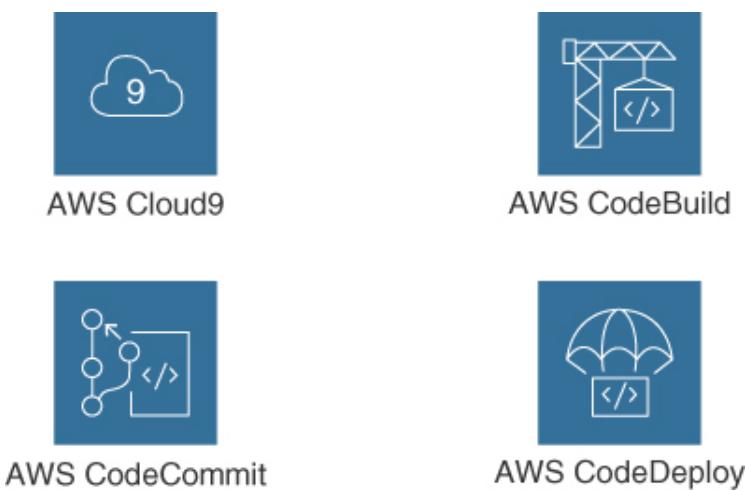
An additional reality is that one cloud provider's PaaS offering is not necessarily compatible with another cloud provider's PaaS offering. For example, Heroku and Microsoft Azure offer

similar PaaS cloud services for developing applications, but internally, each cloud provider operates in a completely different fashion, with a completely different set of supporting APIs. There is no single standard for defining what PaaS must be. Compatibility issues can begin to reveal themselves at the lower levels of each vendor's proposed solution. RESTful interfaces, manifest file formats, framework configurations, external APIs, and component integration are not necessarily compatible across all cloud vendors.

AWS Elastic Beanstalk is Amazon's cloud service for deploying web applications. The service supports Java, .NET, PHP, Node.js, Python, Ruby, and Go. Code applications can be hosted on Apache, Nginx, Passenger, or IIS web servers, and containerized applications hosted on Docker.

Elastic Beanstalk acts as a managed service that frees customers from having to build out infrastructure configurations. It automatically handles scaling, load balancing, monitoring, capacity provisioning, and application updates. For additional details on AWS Elastic Beanstalk, see [Chapter 6, “Designing Resilient Architecture.”](#) AWS has also recently purchased Cloud9, an AWS-hosted integrated development environment (IDE) that supports more than 40 programming languages.

AWS has several cloud services to assist in developing applications, shown in [Figure 1-9](#), including AWS CodeBuild, AWS CodeCommit, AWS Cloud9, and AWS CodeDeploy, that can be key components in your application deployment workflow at AWS.



**Figure 1-9** Platform Options at AWS

## Operational Benefits of AWS

Operating in the public AWS cloud has certain benefits provided by the previously discussed NIST five essential characteristics. Unlimited access to the many cloud services available at AWS may make it easier than expected to operate and manage workloads in the AWS cloud. Consider the following:

- **Servers:** Underutilized servers in your data center are expensive to run and maintain. Moving applications to the public cloud can reduce the size of your on-premises data center. When you no longer host as many physical servers, your total hosting costs (racking, powering, heating, and cooling) could be lower as well. You also don't have to pay for software licenses at the processor level because you're not responsible for running hypervisor services; that's now Amazon's job. You might think that moving to the AWS cloud means virtualized resources and only virtualization. However, with AWS, you can get an ever-increasing variety of EC2 instances, including dedicated virtual servers or bare-metal servers. Sizes range from a single-core CPU with 512 MB of RAM to hundreds of CPU cores and terabytes of RAM.
- **Storage:** Using cloud storage has huge benefits, including having unlimited amounts of storage. Amazon has shareable file solutions for both Linux and Windows Server workloads. Virtual hard disks are available using Amazon EBS to create the required volumes. Unlimited storage and long-term archive storage are provided by Amazon S3 buckets and S3 Glacier archive storage.
- **Managed cloud services:** The AWS-managed cloud services, outlined in [Table 1-1](#), may be able to replace or complement

existing services and utilities currently used on premises after moving to the AWS cloud.

**Table 1-1** Managed Services at AWS

IT Operation	On Premises	AWS Cloud
Monitoring	Nagios, SolarWinds	CloudWatch monitoring metrics for AWS services monitoring logging data unlimited storage. To monitor and perform analysis stored logs in S3 buckets

## **IT Operation**

Data backup

**On Premises**  
Backup tools such as  
Commvault and  
Veritas NetBackup

**AWS Cloud**  
Many third-party  
vendors such as  
Veritas and  
Commvault offer  
many other options.  
the AWS cloud offers  
compatible storage  
appliances like  
Storage Gateway  
also be integrated  
move on-premises  
data records to  
virtual hardware  
volumes to the cloud  
while located in  
popular cloud regions.  
Backup engines can  
be centrally managed  
the backup process  
data storage at  
at AWS to facilitate  
data recovery.

<b>IT Operation Scale</b>	<b>On Premises</b> Automation for increasing/decreasing the size of each virtual machine's RAM and CPU cores as required	<b>AWS Cloud</b> Use EC2 Auto Scaling to automate virtual machines (EC2 instances) containers dynamically increasing the compute required by applications
---------------------------	---	--

**IT Operation**  
Testing/development

**On Premises**  
Expensive  
provisioning of  
hardware for testing  
and development

**AWS Cloud**  
Provisioning  
resources  
term testing  
incredibly  
inexpensively  
up for the  
Tier enables  
customers  
variety of  
services for  
completely  
charge.

<b>IT Operation</b>	<b>On Premises</b>	<b>AWS Clou</b>
Identity management	Active Directory Domain Services for accessing corporate resources	It is possible to migrate or move from on-premises Active Directory to AWS Directory Services to cloud using AWS Directory Service. Deploy AWS Single Sign-on (SSO) using IAM Center to centrally manage access to private cloud business applications running on AWS or in a third-party cloud.



## Cloud Provider Responsibilities

AWS has published service-level agreements (SLAs) for most AWS cloud services. Each separate SLA lists the desired operational level that AWS will endeavor to meet or exceed.

Current details on the SLAs offered by AWS can be viewed at <https://aws.amazon.com/legal/service-level-agreements>. AWS defines its commitments in each SLA about security, compliance, and overall operations. The challenge is to live up to these agreements when all services fail from time to time. Each cloud service SLA contains details about the acceptable outage times and the responsibility of the cloud provider when outages occur. Each SLA also contains statements about their level of responsibility for events outside the cloud provider's control. SLAs commonly use terms such as "best effort" and "commercially reasonable effort."

AWS is responsible for overall service operation and deployment, service orchestration and overall management of their cloud services, the security of the cloud components, and maintenance of each customer's privacy. A managed services SLA also spells out how a cloud consumer is to carry out business with the cloud provider. Each cloud consumer must fully understand what each cloud service offered provides—that is, exactly what the cloud service will, and will not, do.

Is it acceptable to expect AWS failures from time to time? It is a reality; everything does fail from time to time.

What happens when a key service or component of your workload hosted in the AWS cloud fails? Does a disaster occur, or is the failure manageable? When operating at AWS, customers must design each hosted workload to be able to continue operating as required when cloud services, or compute and storage failures occur. Designing high availability and failover for hosted workloads running at AWS is one of the key concepts of many of the questions on the AWS Certified Solutions Architect – Associate (SAA-C03) exam. Many questions will be based on the concepts of designing with a high availability, failover, and durability mindset. Customers must design workloads to meet the application requirements, considering that cloud services *do* fail from time to time.

All public cloud providers really have the same SLA summarized in nine short words when failures happen: “We are sorry; we will give you a credit.” Here’s another reality check: If your application is down, you might have to *prove* that it was actually down by providing network traces and appropriate documentation that leaves no doubt that it was down because of an AWS cloud issue.

Here’s another further detail to be aware of: If you don’t build redundancy into your workload design, don’t bother asking for a credit. Application designs that have a single EC2 instance

hosting a workload with no failover or high-availability design parameters have no cloud provider SLA protection. AWS expects customers to be serious about their application design. Each customer needs to carefully design, deploy, and maintain each hosted workload based on the business needs and requirements, ensuring that any high availability and failover requirements have been met.

## Security at AWS

As you move to the AWS cloud, you need to consider a number of security factors, including the following:

- **Data security:** The reality is that your data is typically more secure and durable when stored in the public cloud than in on-premises physical servers due to the multiple physical copies of any data records stored in public cloud storage. All storage mediums at AWS can also be easily encrypted with the Advanced Encryption Standard (AES). Amazon EBS volumes—both boot and data volumes—can be encrypted at rest and in transit, using customer master keys provided by AWS or keys provided by the customer. Shared storage services such as Amazon EFS and FSx for Windows File Server can also be encrypted at rest, as can all offered database engines. Amazon S3 buckets are encrypted with

keys provided by the S3 service or the Key Management Service (KMS) shown in [Figure 1-10](#). Data durability provides additional security as all data stored in the AWS cloud is stored in multiple physical locations. For example, each EBS volume has multiple copies replicated within the data center where they are created.

Amazon S3 objects are replicated across at least three separate availability zones within the selected AWS region, producing a very high level of durability.

**Default encryption**

Automatically encrypt new objects stored in this bucket. [Learn more](#)

Server-side encryption

Disable

Enable

Encryption key type

To upload an object with a customer-provided encryption key (SSE-C), use the AWS CLI, AWS SDK, or Amazon S3 REST API.

Amazon S3 key (SSE-S3)  
An encryption key that Amazon S3 creates, manages, and uses for you. [Learn more](#)

AWS Key Management Service key (SSE-KMS)  
An encryption key protected by AWS Key Management Service (AWS KMS). [Learn more](#)

**Figure 1-10** Encrypting S3 Buckets Using S3 Keys or AWS-KMS Managed Keys

- **Data privacy:** Amazon ensures that each AWS account's stored data records remain isolated from other AWS

customers. In addition, data records are always created as a private resource. Each S3 bucket can be shared publicly; however, each customer assumes the responsibility when changing a private S3 bucket to be publicly accessible across the Internet.

- **Data control:** Customers are fully responsible for storing and retrieving their data records stored at AWS. It's the customer's responsibility to define the security and accessibility of all data records stored at AWS.
- **Security controls:** AWS Identity and Access management permission policies can be defined at a very granular level to control access to *all* resources at AWS. Customers can also enable multifactor authentication (MFA) as an additional security control for all IAM users authenticating to AWS, and on S3 buckets when deletion of data records is attempted. Resource policies defining the precise level of security and access can be directly attached to resources such as S3 buckets.

## Network Security at AWS

At AWS, networking is managed at the subnet level, and subnets are first created as private subnets with no direct access to the outside world. Subnets that reside on your private networks at AWS are hosted in a virtual private cloud (VPC). Only by adding

gateway services to a VPC and route table entries are subnets able to be accessed from either the Internet, a private VPN connection, or from an external network location. The following are examples of networking services and utilities at AWS that help control network traffic:

- Each subnet's ingress and egress traffic can be controlled by subnet firewalls called *network ACLs* that define separate stateless rules for inbound and outbound packet flow.
- Each EC2 instance hosted on a subnet is protected by a firewall called a *security group*, which defines what inbound traffic is allowed into the instance and where outbound traffic is allowed to flow to.
- VPCs can be further protected by deploying the AWS Network Firewall, providing control over all network traffic, such as blocking outbound Server Message Block (SMB) requests, bad URLs, and specific domain names.
- VPC flow logs can be enabled to capture network traffic for the entire VPC, for a single subnet, or for a network interface.

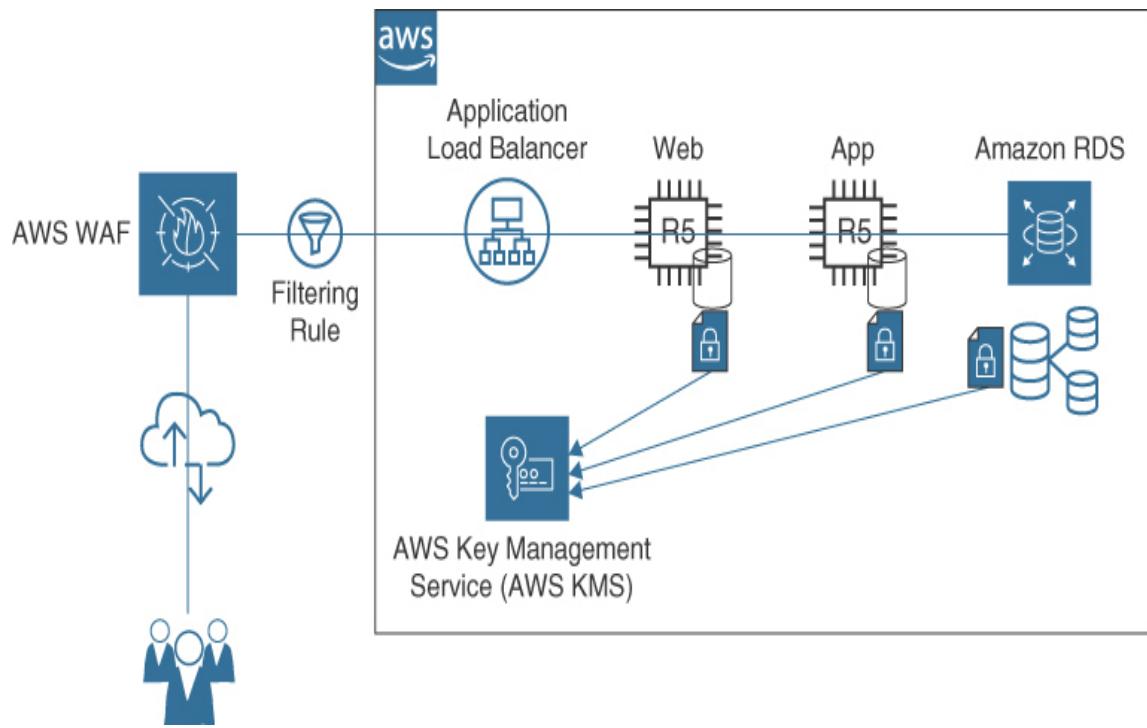
## Application Security at AWS

Both web and application servers hosted at AWS are usually located on private subnets, which are not directly accessible from the Internet. Customers requesting access to the

application will be directed by DNS services (Route 53) to the DNS name of the load balancer, which in turn directs incoming traffic from the public subnet to the targeted web servers hosted in private subnets.

For example, the end-to-end traffic pattern for a three-tier web application can be designed using many encryption/decryption points on its path from source to destination, as described in the list that follows and as shown in [Figure 1-11](#):

- **AWS Web Application Firewall (WAF):** AWS WAF is a custom traffic filter that can be associated with an Application Load Balancer to protect against malicious traffic requests.
- **Application Load Balancer:** An application load balancer can accept encrypted HTTPS traffic on port 443 and provide Secure Sockets Layer/Transport Layer Security (SSL/TLS) decryption and, optionally, user authentication support.
- **EC2 instance hosting a web application:** EBS boot and data volumes can be encrypted using the AWS KMS service.
- **EC2 instance hosting an application server:** EBS boot and data volumes can be encrypted using the AWS KMS service.
- **RDS database server:** All boot and data volumes can be encrypted using the AWS KMS service.



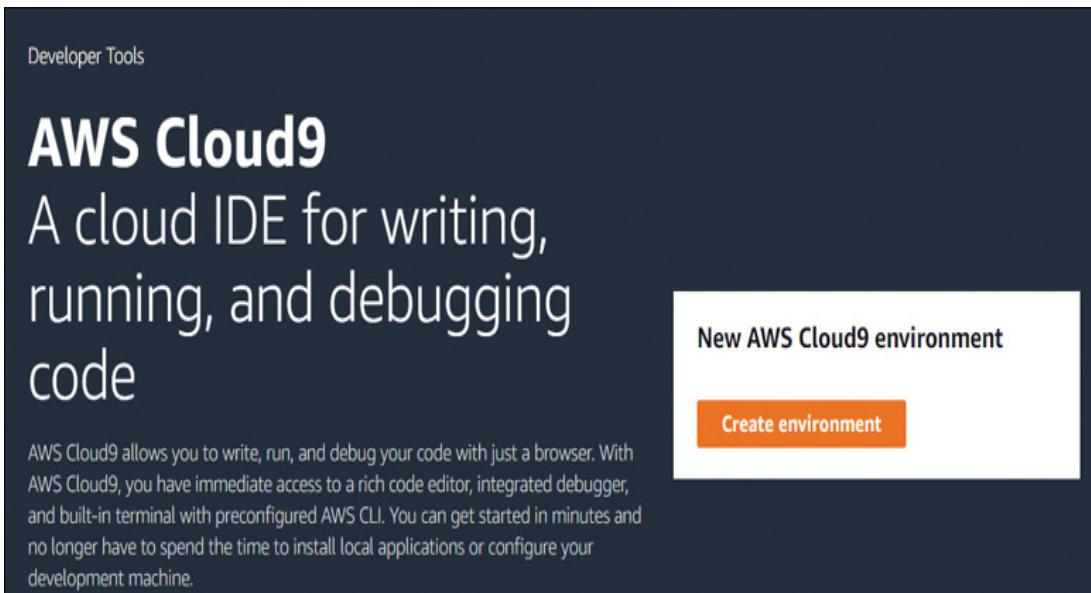
**Figure 1-11** Encrypted Traffic Flow at AWS

## Migrating Applications

For applications that have been chosen as starting candidates to move to the AWS cloud, several decisions need to be made about each application's journey or path. There are several options available for moving an application, depending on factors such as the age of the application and its operating system, and any local dependencies. The following sections walk through these options. Typical large organizations run many applications on thousands of virtual servers. When you move to AWS, you need to determine which applications can be

moved to AWS and what applications should first be prioritized. Consider the following caveats before making these choices:

- **Define a value proposition:** Thousands of companies have successfully moved to AWS; you, too, can be successful. Start off with a defined value proposition that can be validated quickly—that is, in a matter of months rather than years. For developing applications, you could consider developing with AWS Cloud9 (see [Figure 1-12](#)), a cloud-hosted integrated development environment (IDE) that supports more than 40 programming languages. Using Cloud9 and a browser, you can try your hand at developing a new application at AWS or at another PaaS provider such as Heroku. When you develop a completely new application at AWS, you are not constrained by factors such as the type of database that must be used, the type of programming language that must be used, or the type of compute that must be used. Starting new at AWS enables you to try out new methods to host applications, such as serverless computing, creating a mobile application using stateless components, or using DynamoDB as a NoSQL deployment instead of a SQL database. Developing and deploying a new workload at AWS without any legacy dependencies is where the real learning about what the AWS cloud can do for you begins.



**Figure 1-12** Cloud9 IDE at AWS for Application Development

- **Start with low value/low risk:** When choosing what application to move to the AWS cloud, many consultants begin by suggesting a starting point of selecting an already virtualized application stack with high value and low risk. However, it's probably going to take you many months or longer to successfully move a production application to the cloud. Think about choosing an application with low value first. This will enable you to do some additional planning and analysis without any added pressure. Many companies make the pronouncement that applications will be moving to the cloud quickly. It rarely happens as quickly as expected because there are so many things to learn and consider. Take your time and select a working application that has been

virtualized and running successfully. Consider using the AWS Application Migration Service to migrate your first application to AWS. After you are successful, document every step, including lessons learned and what to do differently for the next application chosen to be migrated. Moving additional applications to the cloud will generally be easier and faster thanks to the lessons learned and experience gained.

- **Solve a single problem:** Do you require additional storage? Perhaps that's a great starting point for moving resources to the AWS cloud. Archiving files in S3 Glacier could be as simple as ordering an external AWS Snowball device, connecting it up to your network, filling it with files that you would like to archive, and shipping it back to AWS. Archiving records in the AWS cloud would be an excellent first project in working with AWS.
- **Allowing access to on-premises data records:** The number-one problem for larger companies starting to work with cloud providers is working through the internal politics to allow access to on-premises data from the cloud. Be sure to consider data record access and the steps required for successful access before you begin moving to the cloud:
  - How can you access your on-premises data from the cloud?

- What data records must stay on premises?
- Are you bound by any compliance rules and regulations?
- Is your current data in the right format for what you need?

## **Applications That Can Be Moved to AWS and Hosted on an EC2 Instance with No Changes**

An application that fits into this category is referred to as *lift and shift* or *re-hosting*. Server migration tools and database migration tools can carry out these migrations quite effectively. AWS Application Discovery Service helps organizations plan migration projects by gathering information about their on-premises data centers and potentially thousands of workloads. Server utilization data and the mapping of any dependencies are useful first steps in the initial migration process. The collected data can be exported as a CSV file and used to estimate the total cost of ownership (TCO) of running workloads when planning migration to AWS.

AWS Application Migration Service (formally CloudEndure Migration) is the recommended migration service for performing lift-and-shift migrations to AWS because it automatically converts source servers from physical, virtual, or from existing third-party cloud providers to run at AWS. Supported physical servers include VMware vSphere and

Microsoft Hyper-V EC2 instances can also be migrated between AWS regions or between AWS accounts.

However, applications that are lifted and shifted to the cloud are likely to have dependencies and issues that need to be considered before beginning the migration, including the following:

- If the application stores its data in a database, will the database remain on the premises or will it be moved to the cloud? The Database Migration Service can help in migrating many types of on-premises databases to the cloud.
- If the database for the application remains on premises, are there latency issues that need to be considered when communicating with the database? Each AWS site-to-site VPN connection supports a maximum throughput of up to 1.25 Gbps.
- Will a high-speed connection need to be established between the AWS cloud and the database remaining on premises? A high-speed private fiber AWS Direct Connect dedicated connection ranges from 1 to 100 Gbps.
- Are there compliance issues regarding the application data? Does the data have to be encrypted at rest? Does communication with the database need to be encrypted? AWS Artifact, available in the AWS Management console,

provides compliance reports and agreements to review current compliance standards.

- Do users need to authenticate to the application across the corporate network? If so, are federation services required to be deployed at AWS for single sign-on (SSO)? IAM Identity Center provides SSO for multiple AWS accounts and SaaS cloud applications.
- Are there local dependencies installed on the application server that will interfere with the application server's operation in the AWS cloud? AWS Migration Hub Strategy Recommendations can be useful for alerting customers about potential migration conflicts for application migrations.
- Are there licensing considerations for both the operating system and the application when operating in the cloud? AWS License Manager can help track license usage across your environments.

### **Applications with Many Local Dependencies That Cause Problems When Being Moved to the Cloud**

For applications that fit in this category, consider the following:

- Application developers might have to refactor or restructure the source code of the application to take advantage of managed cloud services such as work queues (Amazon

Simple Queue Service [SQS]), auto scaling (EC2 Auto Scaling), or hosted logging services (CloudWatch logs).

- Application developers might be able to take advantage of AWS cloud services by replacing the existing on-premises database with a database hosted in the cloud utilizing Amazon Relational Database Service (Amazon RDS).

## **Replacing an Existing Application with a SaaS Application Hosted by a Public Cloud Provider**

With so many hosted cloud applications available in the public cloud, the odds are close to 100% that there will be an existing application that can replace a current on-premises application.

## **Applications That Should Remain On Premises and Eventually Be Deprecated**

The following applications should not be moved to the cloud but should remain on premises or should be deprecated:

- The application is hosted on legacy hardware that is near end-of-life.
- The application cannot be virtualized.
- The application does not have technical support.
- The application is used by a small number of users.

## The AWS Well-Architected Framework

Several years ago, AWS introduced the Well-Architected Framework to provide guidance to help cloud architects build secure, resilient, and well-performing infrastructure to host their applications. The framework describes recognized best practices developed over time, based on the experience of many AWS customers and AWS technical experts.

The documentation for the Well-Architected Framework (see <https://docs.aws.amazon.com/wellarchitected/latest/framework>) also presents many key questions customers should review. It is useful to discuss these questions with the other technical team members in your company to make key decisions about your infrastructure and workloads to be hosted at AWS. Each workload to be deployed at AWS should be viewed through the lens of the Well-Architected Framework following these six pillars:

- **Operational excellence:** Relates to how best to design, deploy, execute, and monitor applications running at AWS using automated deployment monitoring procedures, continuous improvement, and automated solutions for recovering from failures. Operational excellence questions to consider include:

- How are disruptions to applications handled—manually or automatically?
- How can you analyze the ongoing health of your applications and infrastructure components hosted at AWS?
- **Security:** Relates to how to best design systems that will operate reliably and securely while protecting customer information and data records. Security questions to consider include:
  - How are security credentials and authentication managed at AWS?
  - How are automated procedures secured?
- **Reliability:** Relates to how applications hosted at AWS recover from disruption with minimal downtime and how applications meet escalating demands. Reliability questions to consider include:
  - How do you monitor resources hosted at AWS?
  - How do applications hosted at AWS adapt to changes in demand by end users?
- **Performance efficiency:** Relates to how to use compute resources to meet and maintain your application requirements on an ongoing basis. Should your compute solution change from EC2 instances to containers or

serverless? Performance efficiency questions to consider include:

- Why did you select your database architecture?
- Why did you select your current compute infrastructure?
- **Cost optimization:** Relates to how to design workloads that meet your needs at the lowest price point. Cost optimization questions to consider include:
  - How do you oversee usage and cost?
  - How do you meet cost targets?
  - Are you aware of current data transfer charges based on your AWS designs?
- **Sustainability:** Relates to designing workload deployments that minimize waste. Sustainability questions to consider include:
  - How do you select the most efficient storage and compute?
  - What managed service offerings could reduce current infrastructure deployments?

## The Well-Architected Tool

In the AWS Management Console, you can search and find the AWS Well-Architected Framework tool. This tool, shown in [Figure 1-13](#), provides a framework for documenting your workloads against AWS best practices, as defined in the Well-Architected Framework documentation. For each of the six

pillars, there are many questions to consider before beginning to deploy an application. As questions for each pillar are considered and debated, milestones can be created marking important points about the workload architecture as teams discuss the questions and make changes to their workload design.

The screenshot shows a navigation path at the top: Well-Architected Tool > Workloads > mobile application > AWS Well-Architected Framework > Review workload. Below this, the title "AWS Well-Architected Framework" is displayed, followed by a link "Add a link to your architectural design". A section titled "OPS 1. How do you determine what your priorities are?" includes an "Info" link. A tip states: "Everyone needs to understand their part in enabling business success. Have shared goals in order to set priorities for resources. This will maximize the benefits of your efforts." Under "Select from the following", there are two options: "Evaluate external customer needs" (with an "Info" link) and "Evaluate internal customer needs" (with an "Info" link). The "Evaluate external customer needs" option is checked.

**Figure 1-13** Evaluating Workloads Using the Well-Architected Framework Tool

The Well-Architected Framework tool provides tips and guidance on how to follow the best practices recommended by AWS while carrying out a full architectural review of an actual

workload that you are planning to deploy at AWS. Your team will find that working with the Well-Architected Framework tool is well worth the time invested.

Before your architectural review begins, open the Well Architected Tool and select the AWS region where your application will be hosted, then define the workload and industry type, and whether the workload is in production or a pre-production environment. After all the pertinent questions have been answered, during the review process, the Well-Architected Framework tool helps you identify potential areas of medium and high risk, based on your answers to the questions. The six pillars of design success are also included in the plan for recommended improvements to your initial design decisions (see [Figure 1-14](#)).



**Figure 1-14** Recommended Improvements Using the Well-Architected Framework Tool Review

## AWS Services Cheat Sheet

Each section of the exam domains for the AWS Certified Solutions Architect – Associated (SAA-C03) exam is covered in a separate chapter in this book. You can quickly understand a variety of AWS services that are covered by the exam domains via a short explanation provided by the following list. You can review additional details on each of these services by reading the related FAQs for each service. There might be exam questions about some of these services, and then again, there might not be. For the purposes of preparing for the exam, the following details for these particular AWS services should be

sufficient in answering the test questions that may mention them.

- **AWS AppSync:** A service designed for mobile applications that require user and data synchronization across multiple devices. AWS AppSync supports iOS, Android, and JavaScript (React and Angular). Select data records can be synchronized automatically across multiple devices using the GraphQL query language.
- **Amazon AppFlow:** A hosted integration service for securely exchanging data records, such as events from external SaaS applications such as Salesforce and ServiceNow.
- **Amazon Athena:** A serverless query service that analyzes Amazon S3 data. Queries can be performed in a variety of standards, including CSV, JSON, ORC, Avro, and Parquet. Queries can also be executed in parallel, resulting in extremely high performance.
- **AWS Audit Manager:** Audit Manager's prebuilt frameworks map your AWS resources to industry standards such as CIS AWS Foundations Benchmark, the General Data Protection Regulation (GDPR), and the Payment Card Industry Data Security Standard (PCI DSS).
- **Amazon Comprehend:** A natural language processing (NLP) service that uses machine learning to find meaning and insights in text.

- **Amazon Cognito:** Add mobile user sign-up, sign-in, and access controls to your web and mobile apps using a hosted identity store that supports both social media and enterprise identity federation.
- **Amazon Detective:** Analyze, investigate, and quickly identify the root cause of potential security issues or suspicious activities collecting log data from your AWS resources using machine learning, statistical analysis, and graph theory, ingesting data from AWS CloudTrail logs, Amazon VPC Flow Logs, and Amazon GuardDuty findings.
- **AWS Device Farm:** An application testing service that lets you improve the quality of your web and mobile apps during development by running tests concurrently on multiple desktop browsers and real physical mobile devices hosted at AWS. Device support includes Apple, Google, and Android devices.
- **AWS Data Exchange:** Supports the secure exchange of third-party data files and data tables into AWS. Customers can use the AWS Data Exchange API to copy selected third-party data from AWS Data Exchange into Amazon S3 storage. Data Exchange third-party products include weather, healthcare, data sciences, geospatial and mapping services.
- **AWS Data Pipeline:** Process and move data between different AWS compute and storage services, and from on-

premises siloed data sources, and transfer the results into Amazon S3 buckets, Amazon RDS, Amazon DynamoDB, and Amazon EMR.

- **Amazon EMR:** EMR is a big data platform for data processing, interactive analysis, and machine learning using Apache Spark, Apache Hive, and Presto. Run petabyte-scale analysis much cheaper than traditional on-premises solutions.
- **Amazon Forecast:** Provides accurate time-sensitive forecasts for retail, manufacturing, travel demand, logistics, and web traffic markets.
- **Amazon Fraud Detector:** A managed fraud detector that helps identify potentially fraudulent online activities such as online payment fraud and fake account creation.
- **AWS Glue:** A fully managed extract, transform, and load (ETL) service that helps discover details and properties of data stored in Amazon S3 and Amazon Redshift for analytics, machine learning, and application development. AWS Glue has the following key components:
  - **AWS Glue Data Catalog:** Stores structural and operational metadata, including its table definition, physical location, and the data's historical and business relevance.
  - **Glue Crawlers:** Crawlers are used to scan various data stores populating the AWS Glue Data Catalog with relevant

data statistics.

- **AWS Glue Studio:** Create jobs that extract structured or semi-structured data from a data source.
- **AWS Glue Schema Registry:** Validate and control streaming data using registered schemas for Apache Avro and JSON.
- **AWS Glue DataBrew:** A visual data preparation tool that can be used by data analysts to clean and normalize data for analysis and machine learning.
- **Amazon Kendra:** Highly accurate machine learning enterprise search service for all unstructured data stored in Amazon S3 and Amazon RDS databases.
- **Amazon Kinesis:** Allows customers to connect, process, and analyze real-time streaming data to quickly gather insights to the incoming data flow of information. The use case for Amazon Kinesis is for ingesting, buffering, and processing streaming video, audio applications, logs, website clickstreams, and IoT telemetry data for machine learning, analysis, and storage at any scale.
- **Amazon Kinesis Video Streams:** Developers can use the Kinesis Video Streams SDK to develop applications with connected camera devices, such as phones, drones, and dash cams, to securely stream video to custom real-time or batch-oriented applications running on AWS EC2

instances. The video streams can also be stored and encrypted for further monitoring and analytics.

- **Amazon Kinesis Data Firehose:** Streaming data is collected and delivered in real time to Amazon S3, Amazon Redshift, Amazon Open Search Service, custom HTTP/HTTPS endpoints, and to third-party service providers including Splunk, Datadog, and LogicMonitor. Kinesis Data Firehouse can also be configured to transform data records before the data is stored.
- **Amazon Kinesis Data Streams:** Collect and process gigabytes of streaming data that is generated continuously from thousands of locations such as log files, e-commerce purchases, game player activity, web clickstream data, and social media information. Multiple data streams ingested into Kinesis are sent into custom applications running on EC2 instances, or data stored in a DynamoDB table, Amazon S3 storage, Amazon EMR, or Amazon Redshift.
- **Amazon Lex:** Build conversational interfaces using voice and text powered by Alexa. Speech recognition and language understanding capabilities enable chatbots for applications published to Facebook Messenger, Slack, or Twilio SMS.
- **Amazon Managed Streaming for Apache Kafka (Amazon MSK):** Streaming data can be consumed using a full-managed Apache Kafka and Kafka Connect Clusters hosted at AWS,

allowing Kafka applications and Kafka connectors to run at AWS without requiring expert knowledge in operating Apache Kafka.

- **Amazon Managed Service for Prometheus:** A monitoring and alerting service that collects and accesses performance and operational data from container workloads on AWS and on premises.
- **Amazon Managed Grafana:** Existing Grafana customers can analyze, monitor, and generate alarms on metrics, logs, and traces across AWS accounts, AWS regions, AWS CloudWatch, AWS X-Ray, Amazon Elasticsearch Service, Amazon Timestream, AWS IoT SiteWise, and Amazon Managed Service for Prometheus.
- **Amazon OpenSearch Service:** Perform log analysis and real-time application monitoring, providing visibility into your workload performance. Find relevant data within applications, websites, and data lakes using SQL query syntax. Data can be read using CSV tables or JSON documents.
- **Amazon Pinpoint:** An outbound and inbound marketing communications service allowing companies to connect with customers using email, SMS, push, voice messages, or in-app messaging to deliver promotional or transactional messages

such as one-time passwords, reminders, or confirmation of orders.

- **Amazon Polly:** Turn text into lifelike speech for speech-enabled mobile apps and devices using lifelike voices in multiple languages; text sent to the Amazon Polly API returns an audio stream for use in your applications or devices.
- **AWS Personal Health Dashboard:** Receive notifications when AWS is experiencing issues on AWS services you are using, and alerts triggered by changes in the health of AWS services.
- **AWS Proton:** Allow platform teams to create rules for developers provisioning automated infrastructure as code.  
There are two supported methods:
  - AWS-managed provisioning uses CloudFormation templates to deploy infrastructure.
  - Self-managed provisioning uses Terraform templates to deploy infrastructure.
- **Amazon QuickSight:** A hosted business intelligence service powered by machine learning that provides data visualizations and insights from an organization's data records for reports or viewable dashboards. Accessed data records can be stored at AWS or stored in external locations including on-premises SQL Server, MySQL, and PostgreSQL

databases, or in Amazon Redshift, RDS, Aurora, Athena, and S3 storage.

- **Amazon Rekognition:** Allows developers to add visual capabilities to applications using the following methods:
  - **Rekognition Image:** Searches, verifies, and organizes millions of images, detecting objects, scenes, and faces; identifies and extracts inappropriate content in images.
  - **Rekognition Video:** Extracts motion-based context from stored or live-stream videos for analysis, recognizing objects, celebrities, and inappropriate content in videos stored in Amazon S3 storage.
- **AWS Security Hub:** Provides a detailed view of your current security environment of a single or multiple AWS accounts by consuming and prioritizing the findings gathered from various AWS security services such as Amazon GuardDuty, AWS Config, Amazon Detective, AWS Firewall Manager, AWS IAM Access Analyzer, Amazon Inspector, Amazon Macie, and Amazon Trusted Advisor. Once enabled, AWS Security Hub executes continuous account-level configuration and security checks based on AWS best practices and industry standards.
- **Amazon SageMaker:** Build, train, and deploy machine learning (ML) models.
  - **Amazon SageMaker Autopilot:** Automatically inspect raw data and apply feature processors picking the best

algorithm training and tuning multiple models and ranking each model based on performance.

- **Amazon SageMaker Pipelines:** Create fully automated ML workflows.
- **Amazon Textract:** A document analysis service that detects and extracts printed text and handwriting from images and scans of uploaded documents.
- **Amazon Transcribe:** Converts speech to text.
- **Amazon Translate:** A neural machine translation service that delivers high-quality language translation.
- **AWS X-Ray:** Allows developers to analyze and debug applications in development and production to quickly identify and troubleshoot performance issues and errors, providing an end-to-end view of workload communication.
  - **Service map:** X-Ray creates a map of services and connections being used by your application and tracks all application requests.
  - **Identify:** Errors and bugs are highlighted by analyzing the response code for each request made to your application.
  - **Custom analysis:** X-Ray query APIs can be used to build your own analysis and visualization interfaces.

## In Conclusion

In this initial chapter, we have looked at what the public cloud is and how AWS fits into the public cloud arena in terms of IaaS and PaaS services. This chapter also introduced the NIST definitions of the public cloud and how the AWS cloud fits into NIST's definition.

This chapter also introduced the AWS Well-Architected Framework, which is an essential guideline on accepted best practices for deploying and managing workloads in the AWS cloud using suggested best practices and procedures. If you are planning to take the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to be familiar with the Well-Architected Framework. We finished with a summary of a variety of AWS services that you might encounter on the exam, with enough details to understand the purpose of each service for answering exam questions.

## Chapter 2

# The AWS Well-Architected Framework

This chapter covers the following topics:

- [The Well-Architected Framework](#)
- [Designing a Workload SLA](#)
- [Deployment Methodologies](#)

This chapter covers content that's important to the following exam domain and task statements:

Domain 2: Design Resilient Architectures

Task Statement 1: Design scalable and loosely coupled architectures

Task Statement 2: Design highly available and/or fault-tolerant architectures

Your organization may be developing applications to be hosted in the cloud, or they may want to move some or all of current IT operations to the cloud. Regardless of the reason or scenario, your organization's applications/workloads that are moved to the cloud will be hosted on a variety of cloud services

maintained and provided by the cloud provider. The most popular public cloud provider competitors to AWS, Microsoft Azure and Google Cloud, have been in operation for well over a decade. During this time, there have been many lessons learned by all cloud providers and customers; what works and what needs to be refined. This learned experience has resulted in many best practices that have been tested and refined for a variety of development and deployment scenarios. Each customer, before moving to the public cloud should take advantage of this documented experience, called the Well-Architected Framework. Microsoft Azure has released the Microsoft Azure Well-Architected Framework (<https://learn.microsoft.com/en-us/azure/architecture/framework/>), and Google has a Google Cloud Architecture Framework (<https://cloud.google.com/architecture/framework>).

The focus of this chapter, and indeed the book, is the pillars of the AWS Well-Architected Framework. The AWS Certified Solutions Architect – Associate website states that “The focus of this certification is on the design of cost and performance optimized solutions, demonstrating a strong understanding of the AWS Well-Architected Framework.”

Selecting the best architectural design for a workload by considering the relevant best practices that have been tested in production by thousands of customers allows organizations to successfully plan for success with a very high degree of confidence. Instead of reacting to a never-ending series of short-term issues and fixes, customers can plan for engineering and operating workloads in the AWS cloud with a proven long-term approach. Whether you have existing systems that you are trying to migrate to the cloud or are building a new project from the ground up, using the AWS Well-Architected Framework will ultimately save your organization a great deal of time by considering many scenarios and ideas for design and deployment that you might not have fully considered.

Keep in mind that you are engineering your workloads for people—your customers. Employees and contractors are building and operating your cloud-based or hybrid deployments, choosing when, where, and how to use the technologies that make sense for each workload deployment. The goal is architecting your operations and workloads to operate successfully in the cloud, meeting and exceeding your business needs and requirements.

## **“Do I Know This Already?”**

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 2-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A](#), “[Answers to the ‘Do I Know This Already?’ Quizzes and Q&A Sections.](#)”

**Table 2-1** “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
The Well-Architected Framework	1, 2
Designing a Workload SLA	3, 4
Deployment Methodologies	5, 6

---

### Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment.

Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

---

**1.** The AWS Well-Architected Framework Sustainability pillar provides guidance on what types of impacts?

1. Reliability
2. Sizing compute and storage to avoid waste
3. Compute performance
4. Storage sizing

**2.** Which other AWS Well-Architected Framework pillars does workload reliability also affect?

1. Cost Optimization and Sustainability
2. Operation Excellence and Sustainability
3. Scale and Performance
4. Security and Cost Optimization

**3.** A workload SLA is designed to meet what criteria?

1. Cloud service SLA
2. Service-level objective
3. Mean time between failures
4. Restore point objective

**4.** What are the metrics used in determining a workload SLA called?

1. Service-level indicators
2. CloudWatch metrics
3. Rules and alerts
4. AWS defined SLAs

**5.** Agile development means focusing on what processes at the same time?

1. Design, coding, and testing
2. Planning and testing
3. Planning and coding
4. Planning, design, coding, and testing

**6.** What companion process can also be used with the AWS Well-Architected Framework?

1. Big Bang development
2. Waterfall development
3. Agile development
4. Twelve-Factor App Methodology

## Foundation Topics

### The Well-Architected Framework

As previously mentioned, AWS, Microsoft Azure, and Google Cloud all have their own well-architected frameworks (WAFs) as guidance, broken up into essential categories. I strongly recommend AWS's guidance for workload deployment (and for preparing for the AWS Certified Solutions Architect – Associate exam; exam questions are based on the Reliability, Security, Performance Efficiency, and Cost Optimization pillars). AWS is in constant contact with its customers to evaluate what they are currently doing in the cloud, and how they are doing it. Customers are always asking for features and changes to be added, and AWS and other public cloud providers are happy to oblige; after all, they want to retain their customers and keep them happy.

New products and services may offer improvements, but only if they are the right fit for your workload and your business

needs and requirements. Decisions should be based on the six pillars of the AWS Well-Architected Framework: Security, Reliability, Performance Efficiency, Cost Optimization, Operational Excellence, and Sustainability. Evaluating these pillars is necessary for both pre-deployment architecture design *and* during your workload solution's lifecycle. Each of the WAF pillars is a subdiscipline of systems engineering in itself. Security engineering, reliability engineering, operational engineering, performance engineering, and cost and sustainability engineering are all areas of concern that customers need to keep on top of.

The AWS Well-Architected Framework has a number of relevant questions for each pillar that each customer should consider (see [Figure 2-1](#)). The end goal is to help you understand both the pros and cons of any decisions that you make when architecting workload successfully in the AWS cloud. The Well-Architected Framework contains best practices for designing reliable, secure, efficient, and cost-effective systems; however, you must carefully consider each best practice offered to see whether it applies. Listed best practices are suggestions, not decrees; final decisions are always left to each organization.

The screenshot shows the AWS Well-Architected Framework interface. On the left, a sidebar lists five questions under the 'Security' pillar: SEC 1. How do you manage credentials and authentication? (Done), SEC 2. How do you control human access?, SEC 3. How do you control programmatic access?, SEC 4. How do you detect and investigate security events?, and SEC 5. How do you defend against emerging security threats?. The main content area is titled 'AWS Well-Architected Framework' and shows the details for SEC 1. It includes a sub-section 'Add a link to your architectural design', a question 'SEC 1. How do you manage credentials and authentication?' with an 'Info' link, and an 'Ask an expert' button. Below this, there is a note about credentials and authentication mechanisms, a radio button for 'Question does not apply to this workload' (selected), and a list of four items: 'Define identity and access management requirements' (Info), 'Secure AWS root user' (Info), 'Enforce use of multi-factor authentication' (Info), and 'Automate enforcement of access controls' (Info) (selected). A navigation bar at the top indicates the path: Well-Architected Tool > Workloads > 2 tier app > AWS Well-Architected Framework > Review workload.

**Figure 2-1** AWS Well-Architected Framework Security Pillar Questions

Before AWS launches a new cloud service, a focus on the desired operational excellence of the proposed service is discussed based on the overall requirements and priorities for the new service and the required business outcomes. There are certain aspects of operational excellence that are considered at the very start of the prepare phase, and there are continual tasks performed during the operation and management of the workload during its lifetime. However, one common thread is woven throughout the operational excellence pillar: performing detailed monitoring of all aspects of each workload during the prepare, operate, and evolve phases. There are four best

practice areas for achieving and maintaining operational excellence:

- **Organization:** Completely understand the entire workload, team responsibilities, and defined business goals for business success for the new workload.
- **Prepare:** Understand the workload to be built, and closely monitor expected behaviors of the workload's internal state and external components.
- **Operate:** Each workload's success is measured by achieving both business and customer outcomes.
- **Evolve:** Continuously improve operations over time through learning and sharing the knowledge learned across all technical teams.

After operation excellence has been addressed, the next goal is to make each workload as secure as possible. Once security at all layers has been considered and planned for, workload reliability is next addressed. Next up is the performance efficiency of the workload; performance should be as fast as required, based on the business requirements. How do you know when there is a security, reliability, performance, or operational issue? To paraphrase: “There are known knowns; there are things we know we know. We also know there are known unknowns; that is to say we know there are some things

we do not know. But there are also unknown unknowns—the ones we don't know we don't know.” Ultimately, the answer to what we don't know always comes back to proactive monitoring. Without monitoring at every level of the workload, technical teams will not be able to monitor the operational health of each workload and won't have enough information to be able to solve existing and unexpected problems.

Let's look at the goals of each pillar of the AWS Well-Architected Framework in brief.



## Operational Excellence Pillar

Operational excellence isn't a one-time effort that you achieve and are done with. Operational excellence is the repeated attempts to achieve greater outcomes from the technologies you choose and to improve your workload's operational models, procedures, principles, patterns, and practices.

Once security, reliability, performance, and cost have been addressed, you will have hopefully achieved a measure of operational excellence for a period of time, perhaps six months, perhaps longer. Then AWS will introduce a new feature, or a

new service that looks interesting; this will send you back into the testing and development phase once again. Perhaps you will find that one of the recently introduced new features will vastly improve how a current workload could be improved. This cycle of change and improvement will continue, forever. Take any new or improved features into account. Operational excellence guidance helps workloads successfully operate in the AWS cloud for the long term.

Operational excellence has been achieved when your workload is operating with just the right amount of security and reliability; the required performance is perfect for your current needs; there is no waste or underutilized components in your application stack; and the cost of running your application is exactly right. Achieving this rarefied level of operation might seem like a fantasy, but in fact it's the ultimate goal of governance processes that have been designed by a Cloud Center of Excellence (Cloud CoE) team within an organization that is tasked with overseeing cloud operations for the organization. The Cloud CoE, and the culture of operational excellence it implements, is the driver that propels improvements and value throughout your organization. But getting there takes work, planning, analysis, and a willingness to make changes to continuously improve and refine operations as a whole.

Changes and improvements to security, reliability, performance efficiency, and cost optimization within your cloud architectures are best communicated through a Cloud CoE. It's also important to realize that changes that affect one pillar might have side effects in other pillars. Changes in security will affect reliability. Changes in improving reliability will affect cost. Operational excellence is where you can review the entire workload and achieve the desired balance.

---

### Note

Operational Excellence design principles include organize, prepare, operate, and evolve. These best practices are achieved through automated processes for operations and deployments, daily and weekly maintenance, and mitigating security incidents when they occur.

---



### Security Pillar

After the initial operational excellence design discussions, implementing a strong security foundation is next. After a

workload has been deployed, the management of workload security is paramount. Organizations must design security into their cloud solution architectures to protect the workload and ensure its survival from attacks and catastrophic events.

If your online store or customer portal is knocked offline by hackers or corrupted with false or damaging information, your organization's reputation and customers could be maligned. What if customers' financial, medical, or other personal information is leaked from your systems? Organizations must design, deploy, test, monitor, and continuously improve security controls from the beginning. Security requires planning, effort, and expense, as nothing is more valuable than your customers and your ability to securely deliver applications and services meeting their needs. Strategies need to be employed to help achieve the security, privacy, and compliance required by each application and its associated components. These strategies are discussed next.

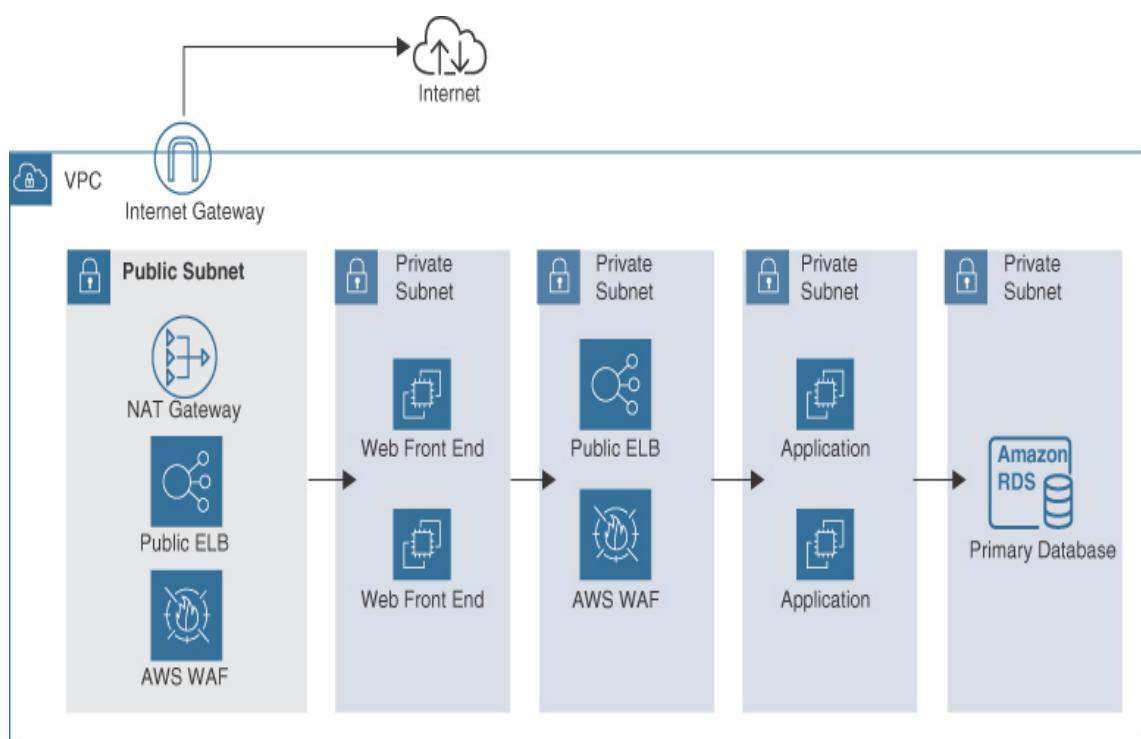


## Defense in Depth

**Defense in depth** can be divided into three areas: physical controls, technical controls, and administrative controls. AWS as the cloud provider is responsible for security of the cloud; therefore, AWS is responsible for securing the physical resources using a variety of methods and controls. AWS also is responsible for providing technical and administrative controls for the cloud services its customers are using, so that they may secure and protect their application stacks and resources that are hosted in the cloud.

Each component of your application stack should have relevant security controls enabled to limit access. For example, consider a two-tier application consisting of web servers and an associated relational database. Both the web and database servers should be hosted on private subnets with no direct access to the Internet. Access to the Internet for updates and licensing should be controlled by using network address translation (NAT) services that allow indirect access from private subnets to the Internet for server updates. For public-facing applications, the load balancer should be placed on a public subnet accepting and directing requests to the web servers hosted on private subnets. Firewalls should be in place at each tier: To protect incoming traffic from Internet attacks, web application firewalls filter out undesirable traffic (see [Figure 2-2](#)). Each web and database server should be also

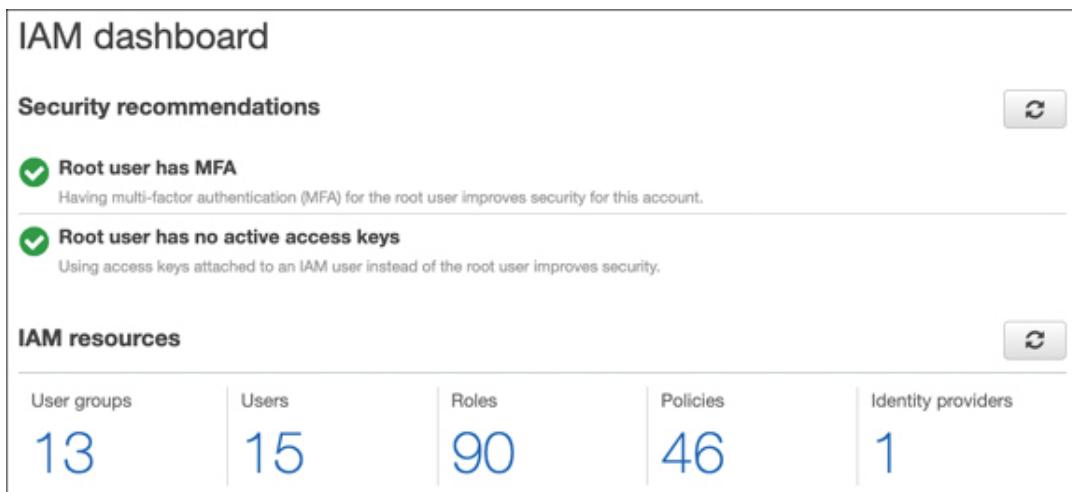
protected by firewalls that allow only the required traffic through. Each subnet should be secured with network access controls that allow the required traffic and deny all other requests. Encryption should be deployed for both data in transit and data at rest.



**Figure 2-2** Defense in Depth Using AWS Services

Other security strategies include implementing the principle of least privilege using identity and authorization controls. AWS Identity and Access Management (IAM) allows customers to create permission policies for users, groups, and roles for cloud

administrators, cloud services, and end users that access cloud resources from their mobile devices (see [Figure 2-3](#)).



**Figure 2-3** Identity and Access Management Security Controls

There are many security services you can enable in the cloud, allowing organizations to reap the benefits. For example, Amazon GuardDuty provides intelligent threat detection using machine learning to provide continuous monitoring of network traffic, DNS queries, and API calls, and protect access to RDS databases and Kubernetes deployments.



## Reliability Pillar

Reliability is the most important requirement; without reliability, end users will eventually stop using the application. Each workload should be designed to minimize failure, keeping in mind there are many components in each application stack to consider. Over the past decade, many best practices have been published as to how to best deploy and manage workloads and associated AWS cloud services with a high degree of reliability.

Organizations must define the required level of reliability for each workload deployed in the AWS cloud. Cloud reliability is commonly defined as cloud service availability. Before the cloud, a ***service-level agreement (SLA)*** was an explicit contract with the provider that included consequences for missing the provider's service-level objectives. AWS SLAs indicate that they will do their best to keep their infrastructure and managed services up and available most of the time. Each AWS cloud service typically has a defined SLA that defines an operational uptime goal which AWS attempts to meet, and usually exceeds (see [Figure 2-4](#)). If failures occur, and they do occur from time to time, and you can prove that a workload was down because of AWS's failures, AWS will provide you with a credit on your bill. AWS SLAs can be found here:  
<https://aws.amazon.com/legal/service-level-agreements/>.

AMAZON ELASTIC COMPUTE CLOUD (EC2)	
Monthly Uptime Percentage	Service Credit Percentage
Less than 99.99% but equal to or greater than 99.0%	10%
Less than 99.0% but equal to or greater than 95.0%	30%
Less than 95.0%	100%

[Amazon Compute Full SLA](#)

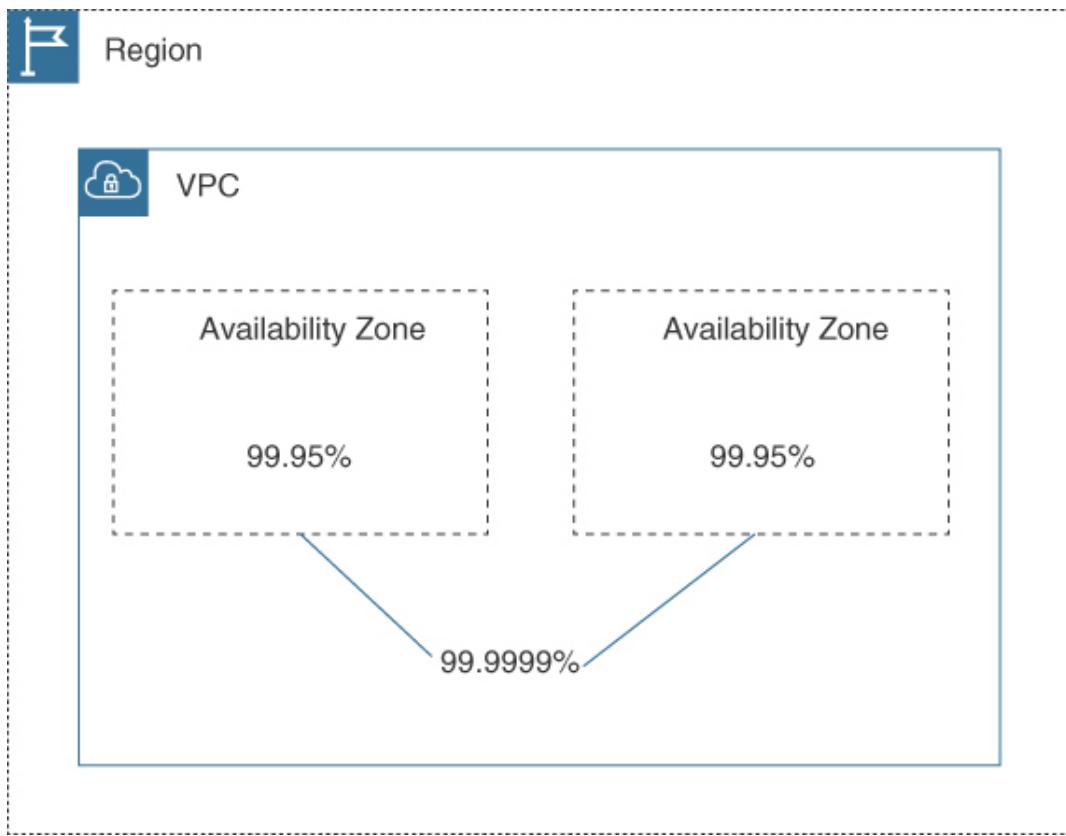
Compute | Containers

**Figure 2-4** AWS Service-Level Agreements

SLA numbers that define cloud service availability look better than they really are. For example, demanding a desired application availability of 99.99% is designing for the potential unavailability over a calendar year of roughly 52 minutes of downtime. Because potential downtime does not include scheduled maintenance, when is this 52 minutes of downtime going to occur? That is the big question to which there is no guaranteed answer. If your workload is streaming video delivery, 99.99% is the recommended availability target to shoot for. If your workload processes ATM transactions, a maximum unavailability of 5 minutes per year, or five nines (99.999%), is the recommended availability target to shoot for. For an online software as a service (SaaS) application involving point-of-sale transactions, the recommendation is 99.95%, or roughly 4 hours

and 22 minutes of downtime per year. Other considerations when calculating workload availability include workload dependencies and availability with redundant components.

- **Workload dependencies:** On-premises workloads will have hard dependencies on other locally installed services; for example, an associated database. Operating in the AWS cloud, if a dependent database fails, a backup or standby database service can be available for automatic failover. In the AWS cloud, workloads can more easily be designed with a reliance on what are defined as *soft dependencies*. With each AWS region containing multiple availability zones, web and application servers deployed across multiple availability zones fronted by a load balancer provide a highly available design. Databases are also deployed across at least two availability zones, and the primary and secondary database servers are kept up to date with synchronous replication.
- **Availability using redundant components:** Designing with independent and redundant cloud services, availability can be calculated by subtracting the availability of the independent cloud services utilized by your workload from 100%. Operating a workload across two availability zones, each independent availability zone has a defined availability of 99.95%. Multiplied together and subtracted from 100%, availability is six nines, or 99.9999% (see [Figure 2-5](#)).



**Figure 2-5** Multi-AZ Service-Level Agreements

---

### Note

Applications that are designed for higher levels of availability will also have increased costs.

Workloads designed with high availability will have multiple web, application, and database instances across multiple availability zones.

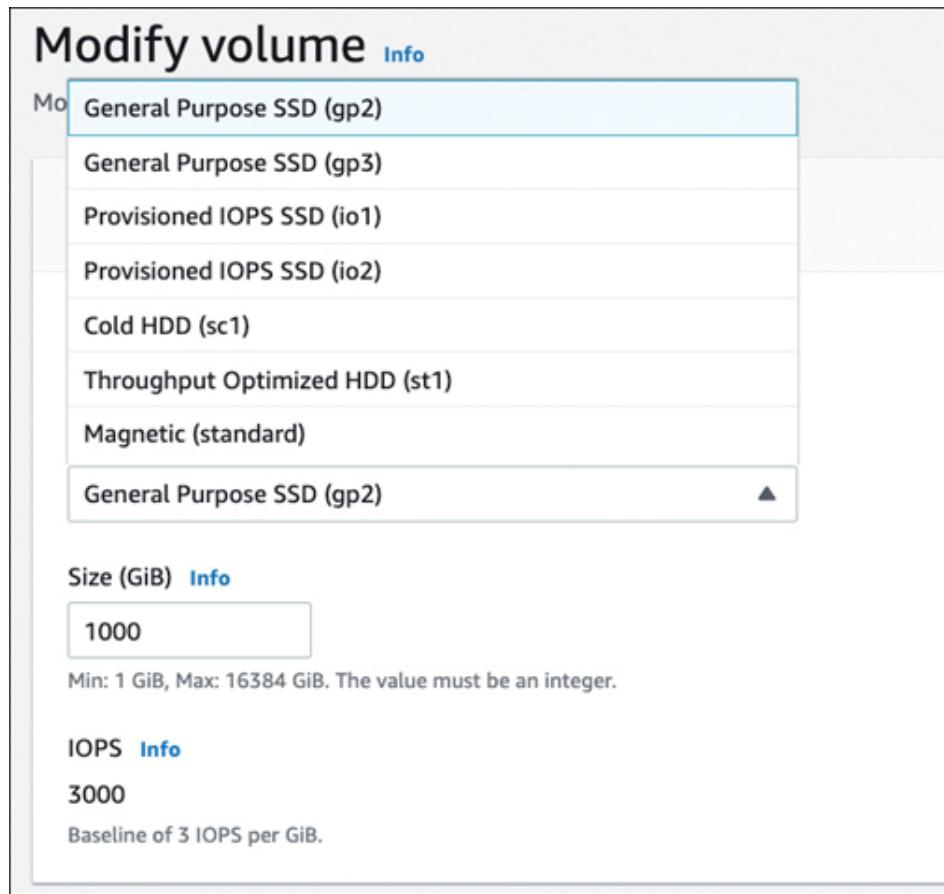
---

## Performance Efficiency Pillar

**Key  
Topic**

In information technology systems engineering, design specialists characterize workloads as compute oriented, storage focused, or memory driven, designing solutions tuned to efficiently meet the design requirements. To achieve your performance goals, customers must address how the design of workload components can affect the performance efficiency of the entire application stack.

How can you design around a compute bottleneck? How do you get around a limit of the read or write rate of your storage? Operating in the AWS cloud, customers can change compute and network performance as simply as turning off their EC2 instances and resizing the EC2 instance, changing the memory, processor, and network specs. Amazon Elastic Block Store (EBS) storage volumes can be changed in size, type, and speed at a moment's notice, increasing volume size and performance as needs change (see [Figure 2-6](#)).



**Figure 2-6** Resize EBS Volumes

Maximizing performance efficiency depends on knowing your workload requirements over time. Measure workload limitations and overall operation by closely monitoring all aspects of the associated cloud services. After several days, weeks, and months of analyzing monitoring data for the computer, networking, and storage services utilized in your application stack, developers and operations teams will be able to make informed decisions about required changes and

improvements to the current workload design. Learn where your bottlenecks are by carefully monitoring all aspects of your application stack. Performance efficiency can also be improved with parallelism, scalability, and improved network speeds:

- Parallelism can be designed into many systems so that you can have many instances running simultaneously. Workload transactions can be serialized using SQS caches or database read replicas.
- Scalability in the cloud is typically horizontal. But scale can also be vertical by increasing the size, compute power, or transaction capacity on web, app, or database processing instances or containers. Customers may find success by increasing the sizes of both the compute and networking speeds of database instances without having to rebuild from scratch, if this is a solution that can be carried out relatively quickly.
- Networking is critical to performance engineering in the cloud, because the host architecture that AWS offers its cloud services on, and the EC2 instances and storage arrays used for all workloads, are all networked. Networking speeds across private networks of AWS can reach 200 Gbps. Connecting to the AWS cloud from an on-premises location privately using VPN connections max out at 1.25 Gbps.

Utilizing high-speed fiber AWS Direct Connections to the AWS cloud range from 1 to 100 Gbps.



## Cost Optimization Pillar

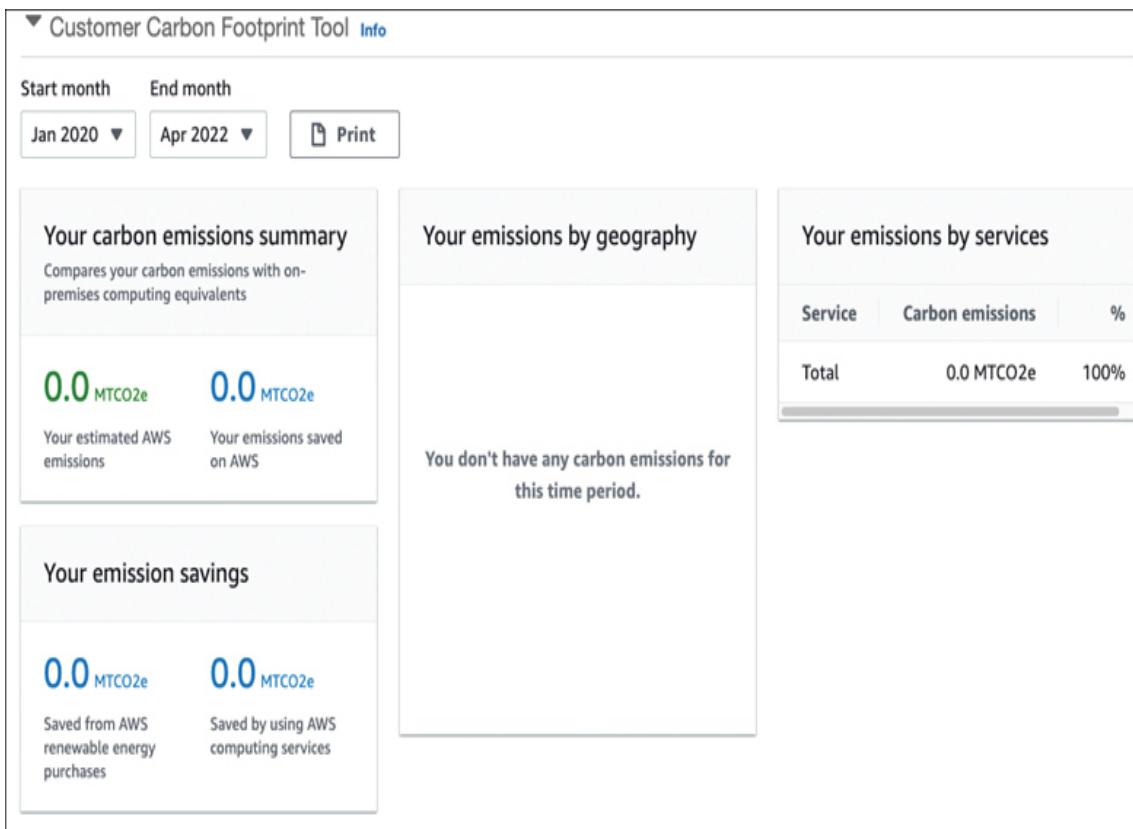
Customers want to optimize their returns on investments in AWS cloud technologies. Successfully reducing cloud costs starts with assessing your true needs. Monitoring is essential to continually improving your performance efficiency, and it's also the key to unlocking cost benefits over time. AWS provides many cost tools for monitoring costs, including cost estimators, trackers, and machine learning-based advisors. Many of these services are free, such as AWS Cost Explorer or AWS Cost and Usage Report.

When analyzing the daily operation of services such as compute and storage, autoscaling and true consumption spending are not the default configuration. Insights into spending trends and rates allow you to control what you spend in the cloud.

## Sustainability Pillar

The Sustainability pillar addresses the impact of your workload deployments against long-term environmental issues such as indirect carbon emissions (see [Figure 2-7](#)) or environmental damage caused by cloud services. Because workload resources rely on cloud services that are virtual compute and storage services, areas where improvements in sustainability include energy consumption and workload efficiency in the following areas:

- Utilizing indirect electricity to power workload resources. Consider deploying resources in AWS regions where the grid has a published carbon intensity lower than other AWS regions.
- Optimizing workloads for economical operations. Consider scaling infrastructure matching user demand and ensuring only the minimum number of resources required are deployed.
- Minimizing the total number of storage resources used by each workload.
- Utilizing managed AWS services instead of existing infrastructure.



**Figure 2-7** Customer Carbon Footprint Tool

## Designing a Workload SLA

Organizations must design workload SLAs that define the level of workload reliability they require and are willing to pay for. It should be noted that AWS cloud services are online, very reliable, and up most of the time. The onus is on each AWS customer to design workloads to be able to minimize the effects of any failures of the AWS cloud services used to create application stacks. Note that each cloud service, such as storage

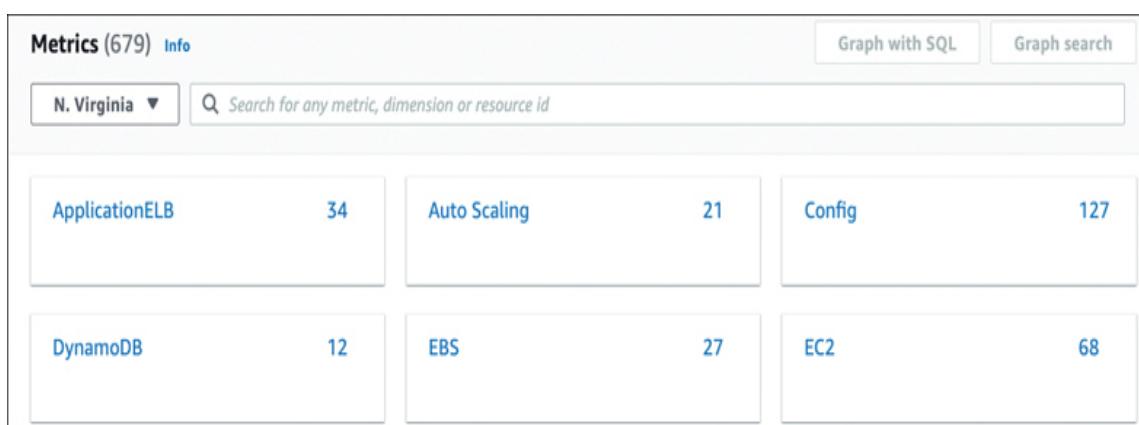
and compute, that is part of your workload application stacks has its own separately defined SLA.

The AWS cloud services that are included in our workloads need to operate at our defined acceptable level of reliability; in the cloud industry this is defined as a **service-level objective (SLO)** and is measured by a metric called a **service-level indicator (SLI)**. For example, web servers must operate at a target of between 55% and 65% utilization. By monitoring the CPU utilization of the web servers, you can be alerted using an Amazon CloudWatch metric linked to an event-driven alarm when the utilization exceeds 65%. You could manually add additional web servers, or EC2 Auto Scaling could be used to automatically add and remove additional web servers as required. There are numerous CloudWatch metrics that can be utilized for more than 70 AWS cloud services providing specific operational details and alert when issues occur.

**Key Topic**

Ongoing CloudWatch monitoring should be used to monitor each integrated cloud service for calculating the reliability of the workload as a whole using the cloud service metrics (see [Figure 2-8](#)). Each metric can be monitored over a defined time

period, which can range from seconds to weeks; the default time period is typically 5 minutes. Every month the average amount of workload availability can be calculated by dividing the successful responses against all requests. By monitoring all integrated cloud services of a workload, valuable knowledge will be gathered regarding reliability issues, potential security threats, and workload performance.



**Figure 2-8** CloudWatch Metrics

Service-level indicators are invaluable for all cloud services; here are a few examples to consider:

- **Availability:** The amount of time that the service is available and usable
- **Latency:** How quickly requests can be fulfilled
- **Throughput:** How much data is being processed; input/output operations per second (IOPS)

- **Durability:** The likelihood data written to storage can be retrieved in the future

## Reliability and Performance Are Linked

Safety, testability, quality, maintainability, stability, durability, and availability are all aspects of a workload's overall reliability. Reliability is the critical design consideration. For example, if a workload crashes periodically but reboots and carries on, persistent end users who retry their requests might get their results eventually. But besides the obvious reliability issue, there is also a performance issue. Your workload's effective performance is lower because of the required retries.

An unresponsive website will eventually cause customer dissatisfaction, which negatively impacts trust and the reputation of the application. If it leads prospective or established customers to shop elsewhere, that could result in lost potential business. Maintaining redundant cloud services to improve workload reliability and performance will result in additional operational expense for some cloud services, such as multiple EC2 instances and multiple database instances providing redundant storage.

When designing for reliability, it's important to realize that not all workload dependencies will have the same impact when

they fail. An outage for an application stack designed with some hard dependencies, such as a single primary database with no alternate database as a backup, will obviously cause problems that cannot be ignored when failure occurs. An outage with a soft dependency, such as an alternate database read-replica, will hopefully have no short-term impact on regular workload operation. Workload reliability can also positively affect overall performance. With the use of multiple availability zones utilizing separate physical data centers separated by miles, workloads can easily achieve a level of reliability and high availability as the web servers, and primary and alternate database servers, are hosted in separate physical locations. Database records can be kept up to date using synchronous replication between the primary and alternate database instances or storage locations.

## Disaster Recovery

In addition to defining your availability objectives, you should consider disaster recovery (DR) objectives. How is each workload recovered when disaster occurs? How much data can you afford to lose, and how quickly must you recover? The application's acceptable **recovery time objective (RTO)** and **recovery point objective (RPO)** must be defined and then tested to ensure that the application meets and possibly exceeds

the desired service-level objectives. Both the RTO and RPO for each workload need to be defined by your organization. RTO is the maximum acceptable delay between the interruption of an application and the restoration of service. RPO is the maximum acceptable amount of data loss.

## Placing Cloud Services

It's critical that you choose where each workload component resides and operates. Some cloud services, such as DNS name resolution and traffic routing and content delivery networks (CDNs), are globally distributed across the world. But most AWS cloud services are regional in design. That is, they are *hosted* in one particular geographical location, even if they might be *accessible* globally. Techniques such as replication, redirection, and load balancing allow you to deploy workload cloud services as multi-region architecture.

Exam questions will ask you to consider several options when deciding where each workload and associated cloud services should be located to best meet the needs of the question's scenario: host location, data caching, data replication, load balancing, and failover architecture that is required.

## Data Residency and Compute Locations

Running workloads in the cloud is essentially leasing time on storage and compute power in a cloud provider's data centers. Each cloud provider hosts services in regions throughout the world. For example, Amazon, Google, and Microsoft each host their cloud services somewhere in the state of Virginia, in a region near Tokyo, and in dozens of other regions around the globe.

How do you choose a region to host your services in? The first suggestion is to consider data residency. Do you have compliance guidelines, laws, or underwriters that suggest or dictate that you store your data within a certain country, state, or province? That might be your sole consideration. If data residency isn't strictly mandated or multiple AWS regions don't meet your criteria, you could instead place your data close to your customers or to your own facilities. Those regions might not be the same geography, depending on the nature of your business and markets you serve.

Let's use an example of a fictitious business called Terra Firma based in Winnipeg, Ontario, which is in the middle of Canada. Let's assume that Terra Firma has deployed its customer portal website in the AWS cloud somewhere in the central Canada region, near its offices and the majority of its users. If customers in the central Canada region have sufficiently fast, low-latency

Internet connectivity to this AWS cloud region, their user experience could be adequate with the hosted website portal. Let's look next at whether caching could be a benefit to the Terra Firma website portal.

## Caching Data with CDNs



CDNs serve up temporary copies of data to the client in order to improve effective network performance. Architecturally, the CDN servers are distributed around many global service areas. In the case of modern cloud CDNs, the service area is global; AWS hosts a global CDN world-wide *cache* called Amazon CloudWatch. How can a CDN cache benefit the web portal users?

Without a CDN cache, the web browsers of Terra Firma's central Canadian customers would send requests to the website address hosted at the cloud region hundreds of miles away. The speed of the website hosting and backend storage and databases for the site are a factor in the end-user experience. But the Internet latency from each user's Internet connection to

the AWS cloud region chosen by Terra Firma can significantly contribute to the performance and overall experience.

Let's assume that Terra Firma's AWS cloud provider CloudFront has a CDN point of presence (POP) located in downtown Winnipeg. If Terra Firma's cloud architects and operations staff configured their website to use a CDN, their customers could benefit from a faster user experience. Customer Julian's web browser queries for Terra Firma's web address and receives a response from the CDN POP location in Winnipeg. Julian's browser next sends a request to load the website to that local POP in Winnipeg, which is a few miles away through the local Internet gateway instead of hundreds of miles and several network hops away to the location of the web server.

If Julian is the first user in the last 24 hours to load any of the requested files, the Winnipeg CDN POP won't find the files in its temporary cache. The POP would then relay a web request to the website origin address hosted in the cloud region hundreds of miles away.

When another customer in the Winnipeg region, Jan, visits the web portal site, her browser resolves the site name to the Winnipeg CDN POP just as Julian's had done recently. For each file that is still cached locally, the POP will send back an

immediate local response; without consulting the web server hundreds of miles away, Jan gets a super-fast experience. And the hundreds, thousands, or millions of other customers who visit the Terra Firma web portal can get this performance benefit as well. AWS CloudFront, AWS's CDN, provides hundreds of POPs around the world (see [Figure 2-9](#)). Wherever users are located, there is likely a POP locally or within the same geographical area. Users close to the actual website AWS cloud region can also be serviced by a POP in that cloud region. People far away will get the same benefits once another end user in their area visits the Terra Firma website, which will automatically populate their local POP's cache.



**Figure 2-9** Amazon CloudWatch Edge Locations

---

### Note

CDN caching techniques must be explicitly programmed and configured in your application code and in the Amazon CloudWatch distribution. Deploying a CDN is designed to improve performance and alleviate the load on the origin services, as in our example of a web portal.

---

## Data Replication

Although CloudFront focuses on performance, a CDN provides substantial security, reliability, cost, and operations benefits. But what if your workload design also stores persistent replica copies of your data, resulting in multiple copies of data? Let's look at the benefits of cloud storage and replicated data records.

The applications, cloud services, and data records that make up your workload solutions must be reliable. A single database or website is a single point of failure that could bring down the entire workload. Organizations don't want their business to fail, so they need to architect and engineer systems to avoid any single points of failure. To improve reliability in cloud solutions architectures, customers need to make choices about redundancy, replication, scaling, and failover.

AWS provides multiple replicas of data records that are stored in the cloud. Three copies is a common default, with options for greater or lesser redundancy depending on the storage service chosen. There are additional options for expanding or reducing additional data replication. Storage and replication are not free; customers must pay for the used storage infrastructure consumed per month. Network transfer costs are billed relative to the rate of change and duplicate copies are billed based on

the stored capacity used. Always check with AWS for up-to-date and region-specific pricing of storage, databases, and replication for these services. Deploy the required configurations at AWS to enable the desired redundancy for every database, data lake, and storage requirement:

- Replicate within an availability zone if necessary
- Replicate between availability zones within your primary AWS region
- Replicate between your chosen primary AWS cloud region and one or more secondary regions

Operating in the AWS cloud requires security and replication at all levels. Customers should also consider deploying compute containers, virtual machine instances, or clusters of either to multiple availability zones and across regions to meet their needs.

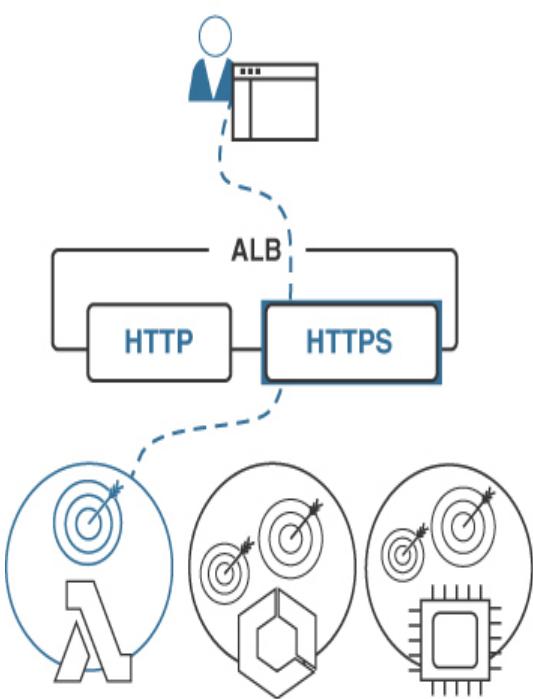
## **Load Balancing Within and Between Regions**

When we have more than one replica of an EC2 instance or containerized application, we can load balance requests to any one of those replicas, thereby balancing the load across the replicas (see [Figure 2-10](#)). Read operations, queries, database

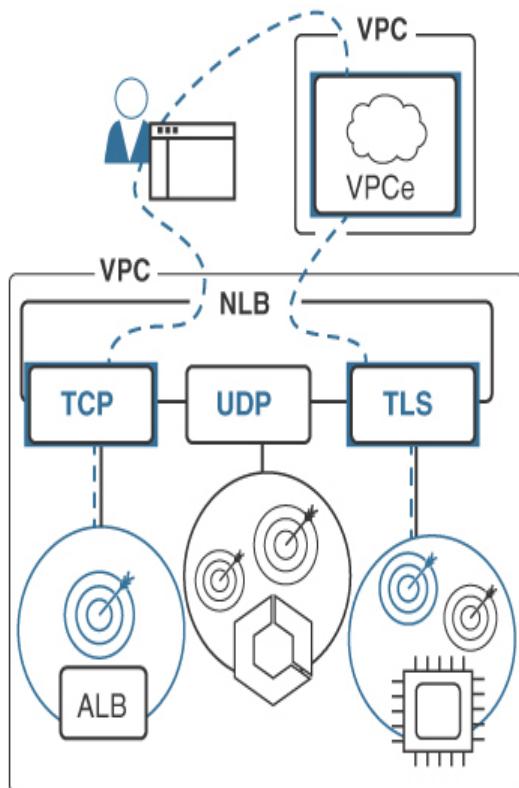
select statements, and many compute requests can be load balanced, as can writes to multi-master database systems.

- Network load balancers use Internet protocol addresses (IPv4 or IPv6), with a choice of TCP, UDP, or both on top of IP and with port numbers to identify the application or service running. Rules stating how to block or relay network traffic matching the source and target IP address, protocol, and port patterns dictate the load-balancing process.
- Application load balancers indicate that HTTP or HTTPS traffic is being load balanced, with balancing rules based on web URLs with rule choices of HTTP or HTTPS, domain name, folder and file name, web path, and query strings.

### Application Load Balancer [Info](#)



### Network Load Balancer [Info](#)



**Figure 2-10** Load Balancer Options

Before load-balancing associations and connections can be established at load balancers, network communications using Internet technologies must perform name resolution. The Domain Name System (DNS) Amazon Route 53 service is used to *resolve* a domain name like [www.pearson.com](http://www.pearson.com) into other domain names (canonically), or into the IPv4 or IPv6 addresses needed to communicate. Rules and policies can be associated with a domain name so that round-robin, priority, failover, weighted, or geolocation will select the results for one end user.

versus another. Amazon Route 53 also provides *inter-region* load balancing across AWS regions.

One of the key advantages of the Network, Application, and Route 53 load balancers is that they can monitor the health of the targets they have been configured to distribute queries to. This feedback enables the load balancers to provide actual *balancing* of the load based on health checks, affinity, stickiness, and security filtering in order to better truly balance the load on your replicas, which allows you to better achieve cost optimization and performance efficiency. Load balancers are also key to application reliability as well.

## **Failover Architecture**

Failover is a critical aspect of load-balancing resources. The ability to have your solutions *detect* failures and *automatically* divert to an alternate replica is essential. With *automatic failover*, you can quickly switchover and reassign requests to the surviving replicas. For database systems, primary–secondary relationships should be designed and deployed.

Failover allows for business continuity in the event of cloud service failure. If there were only two replicas of the application in question and one replica failed, in the failover

state you are running without any redundancy. Customers must restore redundancy by either repairing the failed component and bringing it back online or by replacing it. If you deploy triple or greater degrees of redundancy in the first place, recovery from a failure state is not as immediate a concern.

Failover is a general topic that is not just limited to load-balancing technologies. Additional strategies of failover will be addressed in later chapters. It's important to note that replication, load balancing, and failover are related features that must be orchestrated so that they work in concert with one another. When configured properly, customers can achieve and maintain advantages in security, reliability, and workload performance.

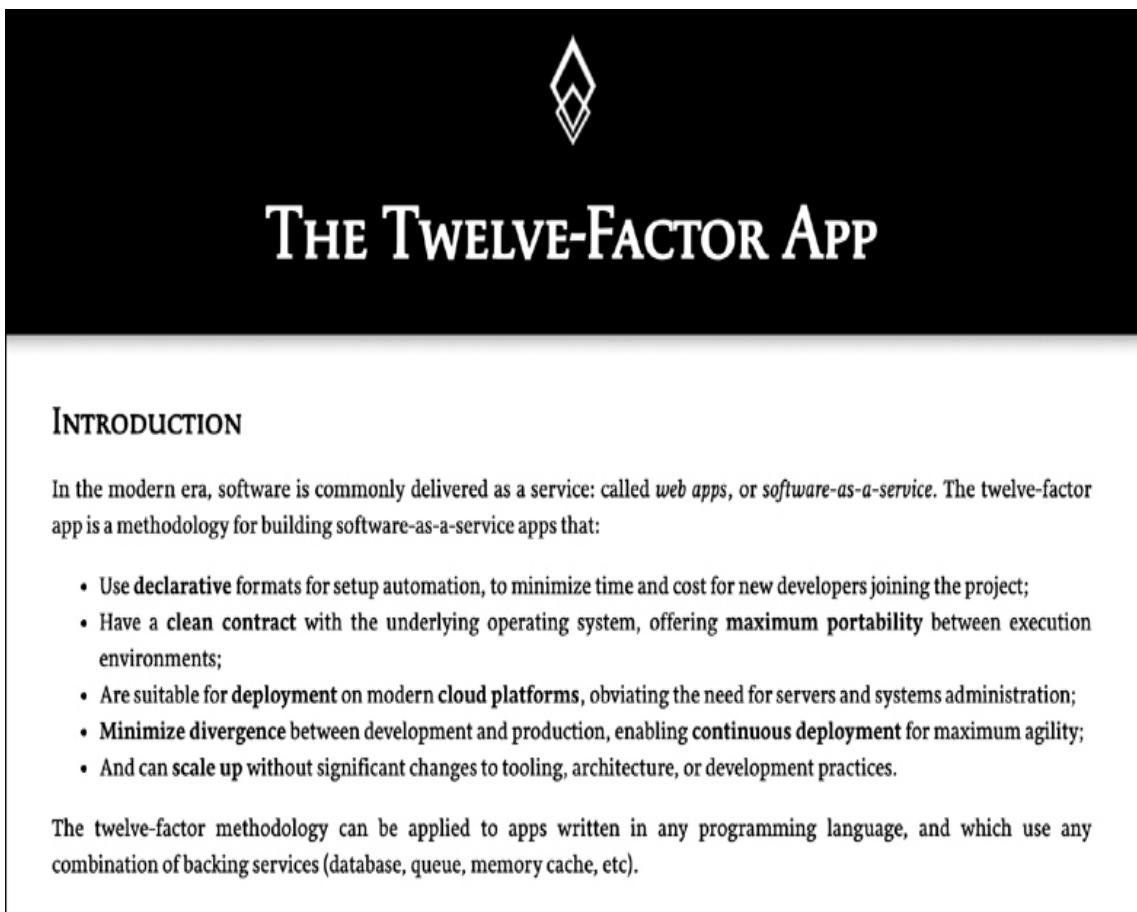
## Deployment Methodologies

Developers getting ready to create their first application in the cloud can look to a number of rules that are generally accepted for successfully creating applications that run exclusively in the public cloud.

Several years ago, Heroku cofounder Adam Wiggins released a suggested blueprint for creating native SaaS applications hosted in the public cloud, called the Twelve-Factor App Methodology.

Heroku (<https://www.heroku.com/>) is a platform as a service provider (PaaS) owned by Salesforce and hosted at AWS.

Heroku was attempting to provide guidance for SaaS applications created in the public cloud based on their real-world experience. Additional details on this methodology can be found at <https://12factor.net/> (see [Figure 2-11](#)).



**Figure 2-11** The 12 Factor App Methodology

These guidelines can be viewed as a set of best practices to consider using when deploying applications at AWS that align with the AWS Well-Architected Framework. Depending on your deployment methods, you may quibble with some of the factors—and that's okay. There are many complementary management services hosted at AWS that greatly speed up the development and deployment process of workloads that are hosted at AWS. The development and operational model that you choose to embrace will follow one of these development and deployment paths:

- **Waterfall:** In this model, deployment is broken down into phases, including proper analysis, system design, implementation and testing, deployment, and ongoing maintenance. In the waterfall model, each of these phases must be completed before the next phase can begin. If your timeline is short, and all the technologies to be used in hosting and managing your workload are fully understood, then perhaps this model can still work in the cloud. However, cloud providers are introducing many cloud services that free you from having to know every technical detail as to how the service works; instead, you can just use the cloud service as part of your workload deployment. For example, an Amazon S3 bucket is used for unlimited cloud storage, without customers needing to know the

technical details of the S3 storage array. In the AWS cloud, all the infrastructure components for storage and compute are already online and functional; you don't have to build storage arrays or even databases from scratch; you can merely order or quickly configure the service, and you are off and running. When developing in the cloud, if your timeline for development is longer than six months, most hosted cloud services will have changed and improved in that time frame, forcing you to take another look at your design and deployment options.

- **Agile:** In this model, the focus is on process adaptability, and teams can be working simultaneously on the planning, design, coding, and testing processes. The entire process cycle is divided into a relatively shorter time frame, such as a 1-month duration. At the end of the first month, the first build of the product is presented to the potential customers, feedback is provided, and is incorporated into the next process cycle and the second version of the product. This process continues until a final production version of the product is delivered and accepted. This process might continue indefinitely if an application has continual changes and updates. Think of any cloud application installed on your personal devices and consider how many times that application gets updated. It's probably updated every few

months at a minimum; for example, the Google Chrome browser updates itself at least every couple of weeks. AWS has a number of cloud services that can help with Agile deployments, including AWS Cloud9, AWS CloudFormation, AWS CodeCommit, AWS CodeBuild, and AWS CodePipeline.

- **Big Bang:** In this model, there is no specific process flow; when money is available in the budget, development starts, and eventually software is developed. This model can work in the cloud because there are no capital costs to worry about; work can proceed when there is a budget. But without proper planning and a full understanding of requirements of the application, long-term projects may have to be constantly revised over time due to changes in the cloud and changes from the customer.

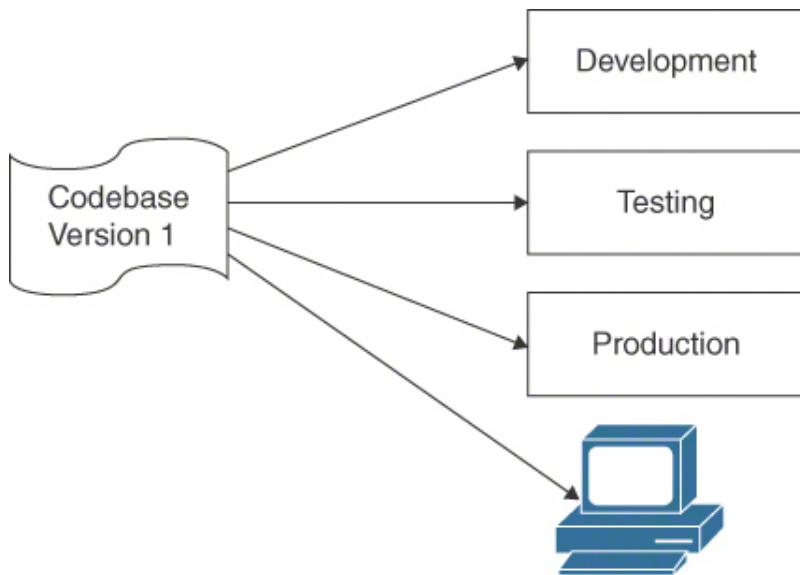
Before deploying applications in the AWS cloud, you should carefully review your current development process and perhaps consider taking some of the steps proposed in the Twelve-Factor App Methodology described in the following sections. Applications that are hosted in the AWS cloud need the correct infrastructure; as a result, the rules for application deployment in the AWS cloud don't stand alone. Cloud infrastructure adhering to the principles of the AWS Well-Architected Framework is also a necessary part of the rules. The following sections look at the applicable factors of the Twelve-

Factor App Methodology from the infrastructure point of view and also identify the AWS services that can help with adhering to each factor. This discussion can help you understand both the factors and the AWS services that are useful in application development and deployment and help you prepare for the exam with the right mindset.

### **Factor 1: Use One Codebase That Is Tracked with Version Control to Allow Many Deployments**

In development circles, this factor is nonnegotiable; it must be followed. Creating an application usually involves three separate environments: development, testing, and production (see [Figure 2-12](#)). The same codebase should be used in each environment, whether it's the developer's laptop, a set of EC2 instances in the test environments, or the production EC2 instances. Each version of application code needs to be stored separately and securely in a safe location. Multiple AWS environments can take advantage of multiple availability zones and multiple VPCs to create dev, test, and production environments.

**Key Topic**



**Figure 2-12** One Codebase, Regardless of Location

Developers typically use code repositories such as GitHub to store their code. Operating systems, off-the-shelf software, dynamic link libraries (DLLs), development environments, and application code are always defined by a specific version number. As your codebase undergoes revisions, each revision of each component needs to be tracked; after all, a single codebase might be responsible for thousands of deployments, and documenting and controlling the separate versions of the codebase just makes sense. Amazon has a code repository, called AWS CodeCommit, for applications developed and hosted at AWS.

At the infrastructure level at Amazon, it is important to consider all dependencies. The AWS infrastructure components

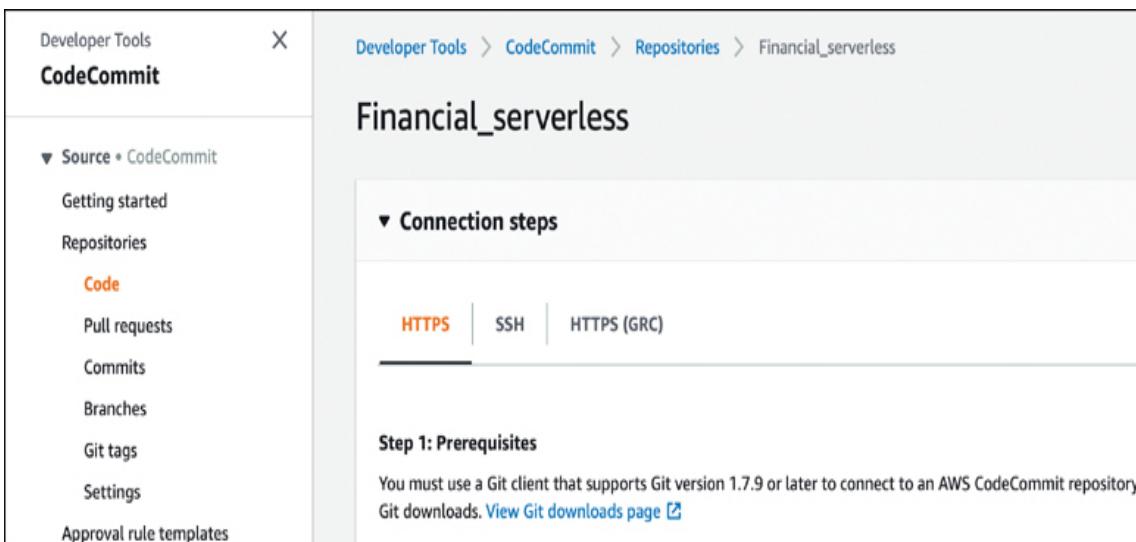
to keep track of include the following:

- **AMIs:** Images for web, application, database, and appliance instances. Amazon Machine Images (AMI) should be version controlled and immutable.
- **Amazon EBS volumes:** Boot volumes and data volumes should be tagged by version number for proper identification and control.
- **Amazon EBS snapshots:** Snapshots used to create virtual server boot volumes are also part of each AMI.
- **Container images:** Each private container image can be stored in the Amazon Elastic Container Registry (ECR) and protected with Identity and Access Management (IAM) permission policies. Container images could be stored in the Amazon Elastic Container Registry.
- **Serverless applications:** The AWS Serverless Application Repository can be used to store, share, and assemble serverless architectures.

## AWS CodeCommit

CodeCommit is a hosted AWS version control service with no storage size limits (see [Figure 2-13](#)). It allows AWS customers to privately store their source code and binary code, which are automatically encrypted at rest and at transit, at AWS.

CodeCommit allows customers to store code versions at AWS rather than at Git without worrying about running out of storage space. CodeCommit is also HIPAA eligible and supports PCI DSS and ISO/IEC 27001 standards.



**Figure 2-13** A CodeCommit Repository

CodeCommit supports common Git commands and, as mentioned earlier, there are no limits on file size, type, and repository size. CodeCommit is designed for collaborative software development environments. When developers make multiple file changes, CodeCommit manages the changes across multiple files. Amazon S3 buckets also support file versioning, but S3 versioning is meant for recovery of older versions of files; it is not designed for collaborative software development

environments; as a result, S3 buckets are better suited for storing files that are not source code.

## Factor 2: Explicitly Declare and Isolate Dependencies

A workload deployed in development, testing, and production VPC networks at AWS may require specific components, such as a MySQL database, a specific operating system version, and a particular utility, and monitoring agent. All dependencies for each workload must be documented so that developers are aware of the version of each component required by the application stack. Deployed applications should never rely on the assumed existence of required system components; ***dependencies*** need to be declared and managed by a dependency manager, ensuring that the required dependencies are installed with the codebase. Examples of dependency managers are Composer, which is used with PHP projects, and Maven, which can be used with Java projects. Dependency managers use a configuration database to keep track of the required version of each component, and what repository to retrieve it from. If there is a specific version of system tools that the codebase always requires, perhaps the system tools could be added to the operating system that the codebase will be installed on. However, over time, software versions for every component will change. The benefit of using a dependency

manager is that the versions of your dependencies will be the same versions used in the development, testing, and production environments.

If multiple operating system versions are deployed, the operating system and its feature set can also be controlled by AMI versions. Several AWS services work with versions of AMIs, application code, and deployment of AWS infrastructure stacks:

- **AWS EC2 Image Builder:** Simplify the building, testing, and deployment of virtual machines and container images at AWS and on premises.
- **AWS CodeCommit:** Can be used to host different versions of the application code.
- **AWS CloudFormation:** Includes several helper scripts to automatically install and configure applications, packages, and operating system services that execute on EC2 Linux and Windows instances. The following are a few examples of these helper scripts:
  - **cfn-init:** This script can install packages, create files, and start operating system services.
  - **cfn-signal:** This script can be used with a wait condition to synchronize installation timings only when the required resources are installed and available.

- **cdn-get-metadata:** This script can be used to retrieve metadata from the EC2 instance's memory.

### Factor 3: Store Configuration in the Environment

Your **codebase** should be the same when it is running in the development, testing, and production network environments. However, your database instances will have different paths, or URLs, when connecting to testing or development environments. Other configuration components, such as API keys, plus database credentials for access and authentication, should never be hard-coded per environment. Use AWS Secrets Manager to store database credentials and secrets. Create IAM roles to access data resources at AWS, including S3 buckets, DynamoDB tables, and RDS databases. You can use Amazon API Gateway to host your APIs.

Development frameworks define environment variables through the use of configuration files. Separating your application components from the application code allows you to reuse your backing services in different environments, using environment variables to point to the desired resource from the development, testing, or production environment. Amazon has a few services that can help centrally store application configurations:

- **AWS Secrets Manager:** This service allows you to store application secrets such as database credentials, API keys, and OAuth tokens.
- **AWS Certificate Manager (ACM):** This service allows you to create and manage public Secure Sockets Layer/Transport Layer Security (SSL/TLS) certificates used for any hosted AWS websites or applications. ACM also enables you to create a private certificate authority and issue X.509 certificates for identification of IAM users, EC2 instances, and AWS services.
- **AWS Key Management Service (AWS KMS):** This service can be used to create and manage encryption keys.
- **AWS CloudHSM:** This service provides single-tenant hardware security modules allowing organizations to generate and manage their own encryption keys at AWS.
- **AWS Systems Manager Parameter Store:** This service stores configuration data and secrets for EC2 instances, including passwords, database strings, and license codes.

#### **Factor 4: Treat Backing Services as Attached Resources**

Many cloud infrastructure and platform services at AWS can be defined as backing services accessed by HTTPS private endpoints connected over the AWS private network. These include databases (for example, Amazon Relational Database Service [RDS], DynamoDB), shared storage (for example,

Amazon S3 buckets, Amazon Elastic File System [EFS]), Simple Mail Transfer Protocol (SMTP) services, queues (for example, Amazon Simple Queue Service [SQS]), caching systems (such as Amazon ElastiCache, which manages Memcached or Redis in-memory queues or in-memory databases), and monitoring services (for example, Amazon CloudWatch, AWS Config, AWS CloudTrail).

Backing services should be completely swappable; for example, a MySQL database hosted on premises should be able to be swapped with a hosted copy of the database at AWS without requiring any changes to application code; for this example, the only variable that would change is the resource handle in the configuration file pointing to the database location.

### **Factor 5: Separate Build and Run Stages**

Applications that will be updated on a defined schedule or at unpredictable times require defined stages during which testing can be carried out on the application state before it is approved and moved into production. AWS Elastic Beanstalk allows you to upload and deploy your application code combined with a configuration file that builds the AWS environment required for the application.

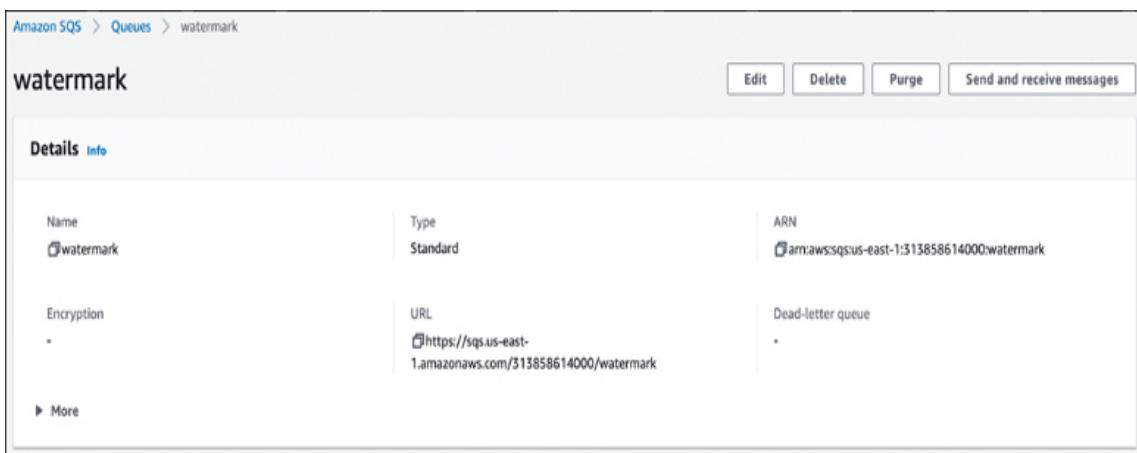
The Elastic Beanstalk build stage could retrieve your application code from a defined repo storage location, such as an Amazon S3 bucket. Developers could also use the Elastic Beanstalk CLI to push application code commits to AWS CodeCommit. When you run the CLI command **EB create** or **EB deploy** to create or update an EBS environment, the selected application version is pulled from the defined AWS CodeCommit repository and the application and required environment are uploaded to Elastic Beanstalk. Other AWS services that work with deployment stages include the following:

- **AWS CodePipeline:** This service provides a continuous delivery service for automating deployment of applications using multiple staging environments.
- **AWS CodeDeploy:** This service helps automate application deployments to EC2 instances hosted at AWS or on premises.
- **AWS CodeBuild:** This service compiles source code, runs tests on prebuilt environments, and produces code ready to deploy without having to manually build the test server environment.

## Factor 6: Execute an App as One or More Stateless Processes

Stateless processes provide fault tolerance for the EC2 instances or containers running applications by separating the application data records and storing them in a centralized storage location such as an Amazon SQS message queue. An example of a stateless design is using an SQS message queue (see [Figure 2-14](#)). EC2 instances that are subscribed to the watermark SQS queue poll the queue, for any updates; when an update message is received, the server carries out the work of adding a watermark to the video and storing the modified video in S3 storage.

### Key Topic



**Figure 2-14** Using SQS Queues to Provide Stateless Memory-Resident Storage for Applications

Other stateless options available at AWS include the following:

- **AWS Simple Notification Service (SNS):** This hosted messaging service allows applications to deliver push-based notifications to subscribers such as SQS queues or Lambda.
- **Amazon MQ:** This hosted managed message broker service, specifically designed for Apache Active MQ, is an open-source message broker service that provides functionality similar to that of AWS SQS queues.
- **Amazon Simple Email Service (SES):** This hosted email-sending service includes an SMTP interface that allows you to integrate the email service into your application for communicating with an end user.
- **AWS Lambda:** This service is used for executing custom functions written by each customer for a vast variety of event-driven tasks. Examples include AWS Config custom rules for resolving infrastructure resources that fall out of compliance, and automated responses for Amazon Simple Notification Services (SNS), and Amazon EventBridge and CloudWatch alarms.
- **Amazon AppFlow:** This service enables you to exchange data between SaaS applications and storage services such as Amazon S3 or Amazon Redshift.
- **AWS Step Functions:** This service enables you to build and run workflows that coordinate the execution of multiple AWS services.

## **Factor 7: Export Services via Port Binding**

Instead of using a local web server installed on a local server host and accessible only from a local port, you should make services accessible by binding to external ports where the services are located and accessible, using an external URL. For example, all web requests can be carried out by binding to an external port, where the web service is hosted and from which it is accessed. The service port that the application needs to connect to is defined by the development environment's configuration file (see the section "[Factor 3: Store Configuration in the Environment](#)," earlier in this chapter. The associated web service can be used multiple times by different applications and the different development, testing, and production environments.



## **Factor 8: Scale Out via the Process Model**

If your application can't scale horizontally, it's not designed for dynamic cloud operation. Many AWS services are designed to automatically scale horizontally:

- **EC2 instances:** Instances and containers can be scaled with EC2 Auto Scaling and CloudWatch metric alarms.
- **Load balancers:** The Elastic Load Balancing (ELB) load balancer infrastructure horizontally scales to handle demand.
  - **Amazon S3 storage:** The S3 storage array infrastructure horizontally scales in the background to handle reads.
  - **Amazon DynamoDB:** The DynamoDB service scales tables within an AWS region. Tables can also be designed as global tables that are asynchronously replicated across multiple AWS regions. Each region table copy is horizontally scaled within the region as required. Each copy of the regional table in each region has a copy of all table data.
- **Amazon Aurora Serverless:** The Amazon Aurora v2 serverless deployment of PostgreSQL or MySQL can be deployed across three availability zones per AWS region, or as a global datastore across multiple AWS regions supporting highly variable workloads.

#### **Factor 9: Maximize Robustness with Fast Startup and Graceful Shutdown**

User session information can be stored in Amazon ElastiCache or in in-memory queues, and application state can be stored in

SQS message queues. Application configuration and bindings, source code, and backing services can be hosted by many AWS-managed services, each with its own levels of redundancy and durability. Data is stored in a persistent storage location such as S3 buckets, RDS databases, DynamoDB databases, or EFS or FSx for Windows File Server shared storage arrays. Workloads with no local dependencies and integrated cloud services can be managed and controlled by a number of AWS management services.

- **Elastic Load Balancer Service:** Load balancers targeting web application hosted on an EC2 instance stop sending requests when ELB health checks fail.
- **Amazon Route 53:** Regional workload failures can be redirected using Route 53 alias records to another region using defined traffic policies.
- **Amazon Relational Database Service (RDS):** When failure occurs, the RDS relational database instances automatically fail over to either the alternate or primary database instance. The failed database instance is automatically rebuilt and brought back online.
- **Amazon DynamoDB:** Tables are replicated across three availability zones throughout each AWS region.
- **EC2 Spot instances:** Spot instances can be configured to automatically hibernate when resources are taken back.

- **Amazon SQS:** SQS messages being processed by EC2 instances that fail are returned to the SQS work queue for reprocessing.
- **AWS Lambda:** Custom function can shut down tagged resources on demand.

## Factor 10: Keep Development, Staging, and Production as Similar as Possible

With this factor, *similar* does not refer to the number of instances or the size of database instances and supporting infrastructure. Your development environment must be exact in the codebase being used but can be dissimilar in the number of instances or database servers being used. Aside from the infrastructure components, everything else in the codebase must remain the same.

- **AWS CloudFormation:** JSON or YAML template files can be used to automatically build AWS infrastructure with conditions that define what infrastructure resources to build for specific development, testing, and production environments.

## Factor 11: Treat Logs as Event Streams

In development, testing, and production environments, each running process log stream must be stored externally. At AWS, logging is designed as event streams.

- **Amazon CloudWatch:** CloudWatch log groups or S3 buckets store EC2 instances' application logs. AWS CloudTrail event logs, which track all API calls to the AWS account, can also be streamed to CloudWatch logs for further analysis.

#### **Factor 12: Run Admin/Management Tasks as One-Off Processes**

Administrative processes should be executed using the same method, regardless of the environment in which the administrative task is executed. For example, an application might require a manual process to be carried out; the steps to carry out the manual process must remain the same, whether they are executed in the development, testing, or production environment.

Several AWS utilities can be used to execute administrative tasks:

- **AWS CLI:** Use the CLI to carry out administrative tasks with scripts.

- **AWS Systems Manager:** Apply OS patches and configure Linux and Windows systems.

## Exam Preparation Tasks

As mentioned in the section “[How to Use This Book](#)” in the Introduction, you have a couple of choices for exam preparation: the exercises here, [Chapter 16](#), “[Final Preparation](#),” and the exam simulation questions in the Pearson Test Prep Software Online.

## Review All Key Topics

Review the most important topics in the chapter, noted with the Key Topic icon in the outer margin of the page. [Table 2-2](#) lists these key topics and the page number on which each is found.



**Table 2-2** [Chapter 2](#) Key Topics

Key Topic Element	Description	Page Number

<b>Key Topic Element</b>	<b>Description</b>	<b>Page Number</b>
Section	Operational Excellence Pillar	44
Section	Security Pillar	45
Section	Defense in Depth	45
Section	Reliability Pillar	47
Section	Performance Efficiency Pillar	49
Section	Cost Optimization Pillar	51
Paragraph	Regular CloudWatch monitoring	53
Section	Caching Data with CDNs	56
<u>Figure 2-12</u>	One Codebase, Regardless of Location	63

<b>Key Topic Element</b>	<b>Description</b>	<b>Page Number</b>
<u>Figure 2-14</u>	Using SQS Queues to Provide Stateless Memory-Resident Storage for Applications	68
Section	Factor 8: Scale Out via the Process Model	69

## Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

defense in depth

service-level agreement (SLA)

service-level objective (SLO)

service-level indicator (SLI)

recovery time objective (RTO)

recovery point objective (RPO)

dependencies

codebase

## Q&A

The answers to these questions appear in [Appendix A](#). For more practice with exam format questions, use the Pearson Test Prep Software Online.

- 1.** Defense in depth can be divided into three areas: physical, technical, and \_\_\_\_\_.
  
- 2.** Application availability of 99.99% means designing for the potential unavailability of roughly 52 minutes of \_\_\_\_\_.
  
- 3.** Determine workload limits by \_\_\_\_\_ all aspects of the application stack.
  
- 4.** Changes in security can affect \_\_\_\_\_.
  
- 5.** Changes in reliability can affect \_\_\_\_\_.
  
- 6.** Availability is defined as the \_\_\_\_\_ of time a cloud service is available and \_\_\_\_\_.

- 7.** If your application can't scale \_\_\_\_\_, it's not designed for \_\_\_\_\_ cloud operation.
- 8.** AWS CloudFormation can be used to automatically build infrastructure using a single \_\_\_\_\_.

# Chapter 3

## Designing Secure Access to AWS Resources

This chapter covers the following topics:

- [Identity and Access Management \(IAM\)](#)
- [AWS IAM Users and Groups](#)
- [Creating IAM Policies](#)
- [IAM Roles](#)
- [AWS Organizations](#)
- [AWS Resource Access Manager](#)
- [AWS Control Tower](#)

This chapter covers content that's important to the following exam domain and task statement:

### Domain 1: Design Secure Architectures

Task Statement 1: Design secure access to AWS resources

It's only natural to think that the same security issues and concerns you face on premises could occur—and possibly be more widespread—when operating in the AWS cloud. Amazon is continuously patching and updating its entire cloud

infrastructure, managed services, and all other integral system components. The latest security bulletins published by AWS (see <https://aws.amazon.com/security/security-bulletins/>) indicate Amazon has needed to consider various security advisories on behalf of all AWS customers.

AWS is responsible for maintaining the security of its cloud infrastructure, defined by AWS as *security of the cloud*. Our job is to maintain everything we host and store in the cloud. This concept is *security in the cloud*.

Indeed, the job of securing customer resources hosted in the AWS cloud never ends, just as security issues continue to be discovered. Way back in 2014, security vulnerabilities with the Bash shell were found. Bash shell security issues weren't new; they just hadn't been discovered until 2014.

To be successful when taking the AWS Certified Solutions Architect – Associate (SAA-C03) exam, a good understanding of the tools and methods available for maintaining the security of workloads, administrators, and cloud services is required.

## **“Do I Know This Already?”**

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or

jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire chapter. [Table 3-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A](#), “[Answers to the ‘Do I Know This Already?’ Quizzes and Q&A Sections.](#)”

**Table 3-1** “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Identity and Access Management (IAM)	1, 2
IAM Users and Groups	3, 4
Creating IAM Policies	5, 6
IAM Roles	7, 8
AWS Organizations	9, 10
AWS Resource Access Manager	11, 12

## Foundation Topics Section

## Questions

AWS Control Tower

13, 14

---

### Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment. Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

---

**1.** What process must happen before AWS IAM grants an IAM user access to requested AWS resources?

1. Authorization
2. Authentication
3. Access granted
4. Access denied

**2.** Which of the following entities is not controlled by AWS IAM?

1. IAM groups
2. IAM user
3. Externally authenticated user
4. Database authentication

**3.** What additional step can be added as a mandatory component when authenticating at AWS?

1. Password policies
2. Multi-factor authentication
3. Resource policies
4. Management policies

**4.** Which of the following AWS IAM entities cannot authenticate?

1. Management policies
2. IAM groups
3. Job policies
4. Secret keys

**5.** What type of permissions policy applies to resources?

1. IAM managed policy

2. Resource policy
3. Job function policy
4. Custom IAM policy

**6.** Which of the following options can be added to a policy as a conditional element?

1. Explicit denies
2. Tags
3. Explicit allows
4. Implicit allows

**7.** What essential component is not attached to an AWS IAM role?

1. Security policy
2. Credentials
3. Multi-factor authentication
4. Tags

**8.** What security service does an AWS IAM role interface with?

1. AWS CloudTrail
2. AWS Security Token Service
3. AWS Config
4. Amazon GuardDuty

**9.** How does an AWS organization help with managing costs?

1. Organizational units
2. Consolidated billing
3. Service control policy
4. Shared security services

**10.** Which of the following is the primary benefit of creating an AWS organization?

1. Lower costs for linked accounts
2. Centralized control of linked AWS accounts
3. Distributed control
4. Nesting OUs

**11.** What is a benefit of using AWS Resource Access Manager without AWS Organizations being deployed?

1. Replacement of network peering
2. Sharing of resources between AWS accounts
3. Sharing of resources between regions
4. Sharing of resources in different availability zones

**12.** Who is in charge of sharing resources using AWS Resource Access Manager?

1. The resource user
2. The principal ID
3. The AWS account root user
4. Any IAM administrator

**13.** How is governance carried out by AWS Control Tower?

1. Account Factory
2. Landing zone
3. Guardrails
4. Dashboard

**14.** What is the purpose of the AWS Control Tower Account Factory?

1. Apply mandatory guardrails
2. Provision new IAM accounts
3. Standardize the provisioning of new AWS accounts
4. Create OUs

## Foundation Topics

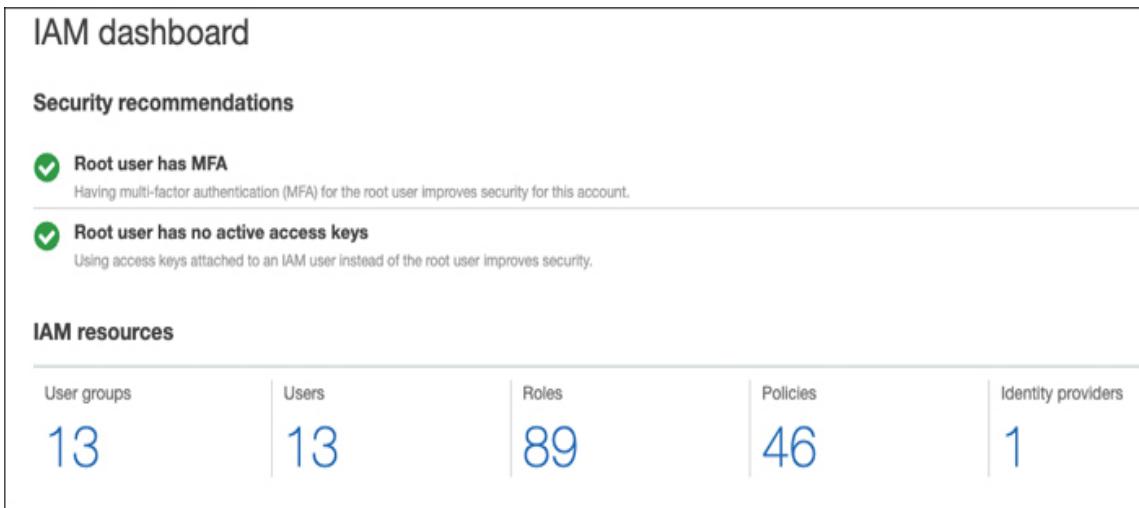
### Identity and Access Management (IAM)

The **Identity and Access Management (IAM)** service is used to deploy and maintain security using security policies and

integrated security services. Security policies define what actions administrators (users and groups) and AWS cloud services can and can't carry out. IAM is a global AWS service that protects AWS resources located around the world across all AWS regions, with a global endpoint located at

<https://sts.amazonaws.com>. There are different user types that require access to AWS resources, including an organization's AWS cloud administrators and end users who are not aware that they are even accessing the AWS cloud when they are using a SaaS application running on their phone. Many popular mobile SaaS applications have backend services hosted at AWS.

Identity and Access Management (IAM) (see [Figure 3-1](#)) was added as a feature to the AWS cloud on September 1, 2010, providing the capability for administrators to define the desired level of access to AWS workloads and the associated cloud services. Using the IAM dashboard, administrators must create IAM users, groups, and roles—no default users, groups, or roles are defined when an AWS account is created.



**Figure 3-1** The IAM Dashboard

IAM is a paranoid security service, and we *want* it to be paranoid. IAM's default mindset is explicit denial with no default access to any AWS cloud service defined.

AWS is responsible for building, securing, and maintaining all the physical components that comprise the cloud services available through the AWS cloud portal. When operating in the public cloud, both the customers and the cloud provider have defined responsibilities, which is defined as a *shared responsibility model*.

The responsibilities of AWS are described as *security of the cloud*—protecting the infrastructure that hosts and runs the AWS cloud services. Each AWS customer's responsibility is defined as *security in the cloud*. Each cloud service ordered by a

customer will have a default security configuration applied, and from this point forward each customer assumes the responsibility of managing and maintaining the cloud service's current and future security configuration. When an EC2 instance is deployed, Amazon is responsible for launching and hosting the EC2 instance on the subnet and availability zone chosen by the customer and ensuring it is initially accessible only to the customer that requested the instance. In the future, a customer may decide to share the EC2 instance with other AWS customers or across the Internet; the exact level of security is the customer's choice—and responsibility. Each AWS service follows the defined shared responsibility model; each party's responsibilities are clearly laid out in AWS documentation. AWS adheres to the ISO/IEC 27001 security standards, which define security management best practices in managing information security through defined security controls.

Amazon cloud services that access the resources in your AWS account on your behalf are also governed by special IAM policies called *service-linked roles* that define the maximum permitted level of access for each cloud service. Identity and Access Management's main features are as follows:

- **Full integration with all AWS services:** Access to every AWS service can be controlled using IAM security.
- **Cost benefits:** There is no additional charge for using IAM security to control access to AWS resources.
- **Controlled access to your AWS account resources in all regions:** Each IAM user can be assigned security policies controlling access to AWS resources in any AWS account in any AWS region.
- **Granular permission control:** IAM can control access to AWS resources to a granular level, for example, defining a policy that allows an IAM user the singular task of viewing a load balancer's attributes.
- **Define the level of access AWS services have to resources:** When you order an AWS cloud service such as AWS Config, the service is granted access to your AWS account through an IAM security policy controlled by a service-linked role (see [Figure 3-2](#)), ensuring that the AWS service can only carry out the approved list of tasks.
- **Multi-factor authentication (MFA):** An additional layer of authentication can be added to any IAM user, including the root user of each AWS account. [\*\*\*Multi-factor authentication \(MFA\)\*\*\*](#) provides a security code—from a software or hardware device that is linked to your AWS account—that

must be entered and verified in order for authentication to AWS to succeed.

- **Identity federation/SSO access to AWS services:** IAM roles allow *externally authenticated users*—whose identities have been federated from a corporate directory or from a third-party identity provider such as Google or Facebook—to have temporary access to select AWS services.



The screenshot shows the AWS Config Settings page. At the top, it says "AWS Config > Settings". Below that is a section titled "Recorder" which has a green checkmark next to "Recording is on". Under "General settings", there are two columns. The left column lists "Resource types to record" with options "Record all resources supported in this region" and "Include global resources (e.g., AWS IAM resources)". The right column shows an "AWS Config role" named "config-role-us-east-1". At the bottom, it shows "Data retention period" with "Default period is 7 years".

**Figure 3-2** AWS Config Service-Linked Roles Defined by IAM

---

### Note

Because IAM is a global service, security changes and additions to the IAM service can take several minutes to completely replicate across the AWS cloud.

---



## IAM Policy Definitions

Each IAM policy is a set of rules that define the actions that each AWS entity can perform on specific AWS resources; *who* is allowed to do *what*. To understand IAM, we need to understand the following terms:

- **User:** Only defined IAM users within each AWS account and externally authenticated users with assigned roles can authenticate to an AWS account. An example of an externally authenticated user could be a corporate user who first authenticates to the corporate Active Directory network that also requires access to AWS resources. After AWS verifies the externally authenticated user's attached IAM role, temporary credentials are assigned, allowing the external authenticated user access to the requested AWS resources. Google and

Facebook users are examples of externally authenticated users supported by IAM roles and AWS.

- **Group:** A group of IAM users can access AWS resources based on the IAM policies assigned to the IAM group they belong to.
- **Policy:** Each AWS service can be controlled by IAM policies created and managed by AWS or by custom policies created by customers.
- **Statement:** Policy statements define what actions are allowed or denied to AWS resources.
- **Principal:** The principal is an IAM user or application that can perform actions on an AWS resource.
- **Resource:** A resource is an AWS resource (such as compute, storage, networking, or managed services) where actions are performed.
- **Identity:** An identity is the IAM user, group, or role where an IAM policy is attached.
- **Entities:** IAM entities that can authenticate are an IAM user, which is assigned permanent credentials, or an IAM role, which does not have attached credentials (that is, no password or permanent access keys). Temporary authentication credentials and session tokens are assigned to a role only after verification confirms that the identity is allowed to assume the policy assigned to the IAM role.

- **Role:** An IAM role provides temporary access to AWS resources based on the attached IAM policy.
- **Condition:** Specific conditions can be mandated. For example, a specific principal, IP address, date, or tag must be present before access is allowed. Conditions are optional.

## IAM Authentication



Before tasks can be performed on AWS resources, you must first be authenticated as an IAM user signing in with a recognized IAM username and password or have been granted access using an IAM Role. If multi-factor authentication is enabled, you must also enter a numerical code during the authentication process before authentication is successful.

Authentication is also required when running commands or scripts using the AWS command-line interface (CLI) or software development kit (SDK). A valid **access key** and secret access key assigned to the IAM user account making the CLI or SDK request must be provided and validated before the command or script will execute.

When an IAM user account is created, two access keys are created; the first access key is the *ID key*, which is an uppercase alphabetic string of characters in the format AKIAXXXXXXXXXXXX, as shown in [Figure 3-3](#). The second access key, the *secret access key*, is a Base64 string that is 40 characters in length.

**Key Topic**

The screenshot shows the 'Add user' success page. It includes a success message, download options, and a user table.

**Success:**  
You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://mbw.signin.aws.amazon.com/console>

**Download .csv**

	User	Access key ID	Secret access key	Password	Email login instructions
<input checked="" type="checkbox"/>	Susan	AKIAUSE3OMLYMV77IB6Z <a href="#">Edit</a>	PG/0hbbDVpYDQDBKB7M GcWFej3uthVxUEuga86R <a href="#">Hide</a>	bo[6-eENRPasx <a href="#">Hide</a>	<a href="#">Send email</a>

**Figure 3-3** IAM User Account Access Keys

External authentication uses a set of temporary access keys issued by the AWS Security Token Service (AWS STS). Temporary access keys issued from STS are in the format

ASIAXXXXXX. External authentication is covered in more detail later in this chapter.

Several forms of authentication are supported by AWS, including the following:

- IAM users and groups carry out administrative actions on AWS services and resources.
- Single sign-on (SSO) is supported using a federated identity with Active Directory Domain Services (AD DS) or a third-party provider that supports the Security Assertion Markup Language (SAML) 2.0 protocol.
- SAML external authentication is supported by IAM using IAM roles, which are attached to the externally authenticated user after identity verification of their external identity by the AWS Security Token Service. Active Directory credentials are stored in two locations: on the Active Directory domain controllers on premises, and on domain controllers hosted at AWS that have been synchronized with a current copy of the organization's Active Directory credentials and attributes.
- Amazon Cognito authenticates and controls access to AWS resources from mobile applications using IAM policies and IAM roles using an identity store and application data synchronization.

- AWS supports external authentication from mobile applications using public identity providers, Facebook, Google, Login with Amazon, and providers that support OpenID Connect, which generates the authentication token that is presented to AWS STS. Verification of the external authentication token by STS results in temporary security credentials being provided for access to the desired AWS resources.
- IAM RDS database authentication is supported by the following database engines:
  - MariaDB 10.6, all minor versions
  - MySQL 8.0, minor version 8.0.23 or higher
  - MySQL 5.7, minor version 5.7.33 or higher
  - PostgreSQL 14, 13, 12, and 11, all minor versions
  - PostgreSQL 10, minor version 10.6 or higher
  - PostgreSQL 9.6, minor version 9.6.11 or higher
  - PostgreSQL 9.5, minor version 9.5.15 or higher

## **Requesting Access to AWS Resources**

Only after authentication is successful are IAM users allowed to request access to AWS resources. The following IAM components work together when requesting access to AWS resources:

- **Principal:** The principal defines which IAM user or external user with an assigned IAM role has requested access.
- **Operations:** Only after each request is authenticated and authorized are operations and actions to the requested AWS resource approved. Operations are always API calls executed from the AWS Management Console, through AWS Lambda function, a CLI command or script, or an AWS SDK using RESTful calls.
- **Actions:** Actions define the specific task, or tasks, the principal has requested to perform. Actions might be for information (**List** or **Get** requests) or to make changes (such as creating, deleting, or modifying).
- **Resource:** Every AWS resource is identified with a unique Amazon Resource Name (ARN), as shown in [Figure 3-4](#).
- **Environmental data:** Environmental data indicates where the request originated (for example, from a specific IP address range) and can provide additional required security information such as the time of day.
- **Resource data:** Resource data provides additional details about the resource being accessed, such as a specific Amazon S3 bucket, Amazon DynamoDB table, or a specific tag attached to the AWS resource being accessed.

## Key Topic

The screenshot shows the 'Summary' tab for a user named 'Susan'. Key details include:

- User ARN: arn:aws:iam::313858614000:user/Susan
- Path: /
- Creation time: 2022-06-17 11:56 EDT

Below the summary, there are tabs for 'Permissions', 'Groups (1)', 'Tags (1)', 'Security credentials', and 'Access Advisor'. The 'Permissions' tab is selected, showing:

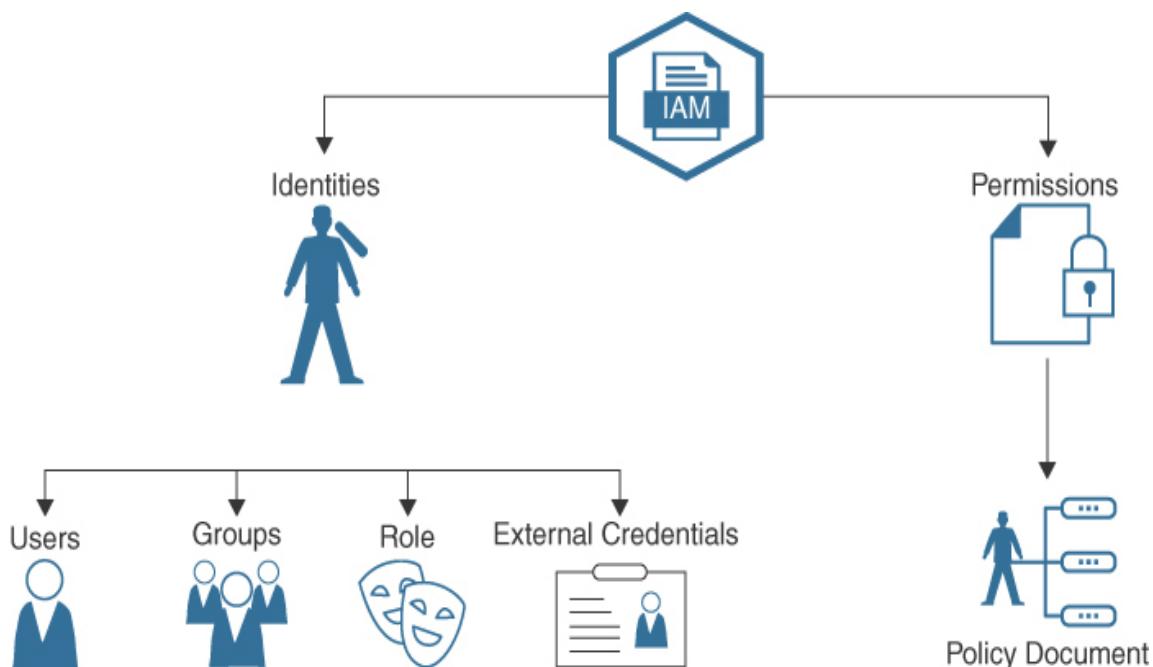
- A dropdown menu for 'Permissions policies (2 policies applied)'
- A blue 'Add permissions' button.
- A table showing attached policies:
  - Policy name: AmazonEC2FullAccess
  - Attached from group: (indicated by a right arrow icon)

**Figure 3-4** An Amazon Resource Name (ARN)

## The Authorization Process

IAM reviews each request against the attached policies of the principal requesting authorization and determines whether the request will be allowed or denied, as shown in [Figure 3-5](#). Note that the principal might also be a member of one or more **IAM groups**, which will increase the number of assigned policies that need to be evaluated before authorization is approved or denied. The evaluation logic of IAM policies follows these strict rules:

- By default, all requests are implicitly denied; there are no implicit permissions. Actions are not allowed without an explicit allow.
- Policies are evaluated for an explicit deny; if found the action is denied.
- An explicit allow overrides the implicit deny, allowing the action to be carried out.
- An explicit deny denies a requested action.



**Figure 3-5 IAM Authorization**

When a principal makes a request to access AWS resources, the IAM service confirms the principal is authenticated, signed in, authorized, and has the necessary permissions.

---

## Note

You probably experienced an IAM-like authorization process as a teenager. It might have sounded like this: “Hey, Dad, can I go to the movies?” “Nope. All requests are denied.” So, you wait until Mom gets home. “Hey, Mom, can I go to the movies?” “I think so, but let me see whether you cleaned your room. Nope. You can’t go to the movies because you didn’t clean your room.” Mom was sneaky and also used a condition; IAM policies can also use conditions for additional control.

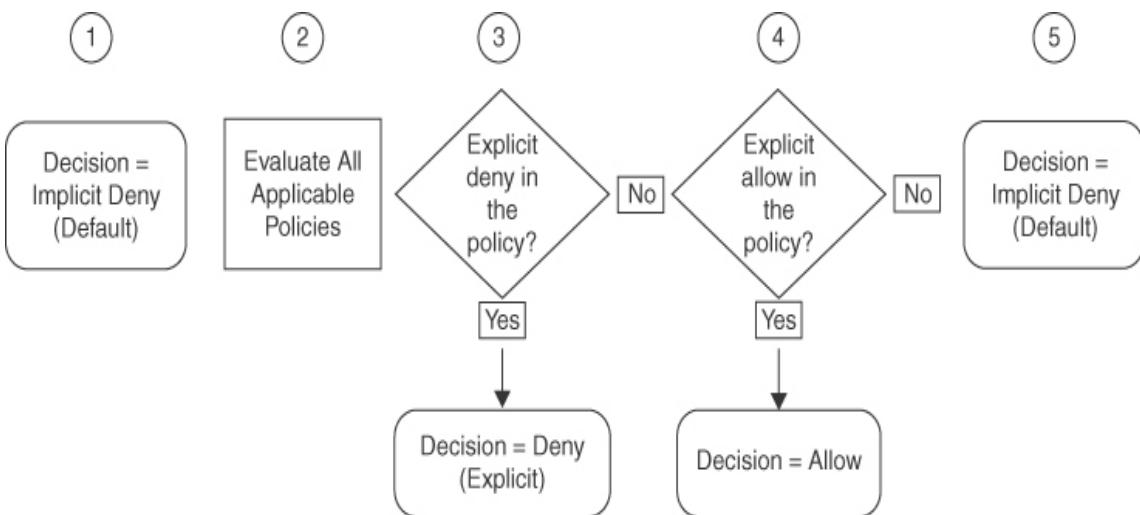
---

The IAM security system reviews the policies assigned and approves or denies the request. As mentioned earlier, IAM implicitly denies everything by default. Requests are authorized only if the specific request is allowed. The following request logic maps to [Figure 3-6](#):

1. The evaluation logic follows exact rules that can’t be bent for anyone, not even Jeff Bezos; implicit deny by default. All requests are implicitly denied by default for IAM users.
2. Attached policies are evaluated.
3. An explicit deny denies any request. A default deny can only be overruled and allowed by an explicit allow permission.

4. Each explicit allow permission in the attached policies is allowed.
5. An explicit deny in any policy results in no access.

## Key Topic



**Figure 3-6** Policy Evaluation Logic

## Actions

Actions are the tasks you can perform on an AWS resource, such as creating, editing, and deleting a resource. There can be many actions for each resource; for example, the EC2 service has more than 400 different actions that can be allowed or denied (see [Figure 3-7](#)). Once specific actions have been

approved, only those actions, which are allowed in the policy, can be performed on the defined resource.

The screenshot shows the 'Actions' section of the IAM policy editor for the EC2 service. At the top, there's a header with 'EC2 (All actions)' and '33 warnings'. Below it, a 'Service' dropdown is set to 'EC2'. On the right, there are 'Clone' and 'Remove' buttons. The main area starts with a 'Actions' heading and a note to 'Specify the actions allowed in EC2'. A 'Filter actions' search bar is present. Under 'Manual actions', 'All EC2 actions (ec2:\*)' is checked. The 'Access level' section lists several options: 'List (119 selected)', 'Read (22 selected)', 'Tagging (2 selected)', 'Write (282 selected)', and 'Permissions management (5 selected)'. To the right of this section are 'Expand all' and 'Collapse all' buttons. At the bottom, there's a 'Action warnings' section with four items, each requiring '1 more action' for full permissions: 'ec2:ReplaceIamInstanceProfileAssociation', 'ec2>CreateFlowLogs', 'ec2>CreateVpcEndpoint', and 'ec2:AssociateIamInstanceProfile'.

**Figure 3-7** Actions Approved by IAM

Each AWS resource has several actions that can be carried out; the initial access level choices are typically **List**, **Read** (typically **Get or List**), and **Write** (typically **Create**, **Put**, **Delete**, **Update**). The type of AWS resource determines the action choices that are available.

## IAM Users and Groups

When you think of the word *account*, you might think specifically of a user account or a group account in the context of an operating system. At AWS, the account that you initially signed up for was designed for organization-wide use, but each AWS account can be used by a single individual, or an organization. It can seem confusing at first.

It might help to think of your AWS account as a complete hosted cloud operating system with security features comparable to the Red Hat Enterprise Linux operating system or Microsoft Active Directory Domain Services.

Many organizations use many AWS accounts—perhaps one or more per developer. At AWS, all cloud services are available per AWS account—subject to the permissions and policies assigned to the authenticating IAM user accounts and roles.

Within each AWS account, IAM user identities or IAM roles are created for these requirements:

- An administrator who needs access to the AWS Management Console.
- An administrator or a developer who needs access to the AWS APIs using the AWS Management Console, and using the

AWS CLI command-line interface typing single commands or running scripts, or development of applications using AWS SDKs, such as JavaScript or .NET.

---

### **Note**

All companies need to consider the number of administrator and developer user accounts that need to be created and the number of AWS accounts that need to be managed. The best practice is to create roles for external access (access to other AWS accounts or federated access) as much as possible. Roles are explained later in this chapter.

---

### **The Root User**

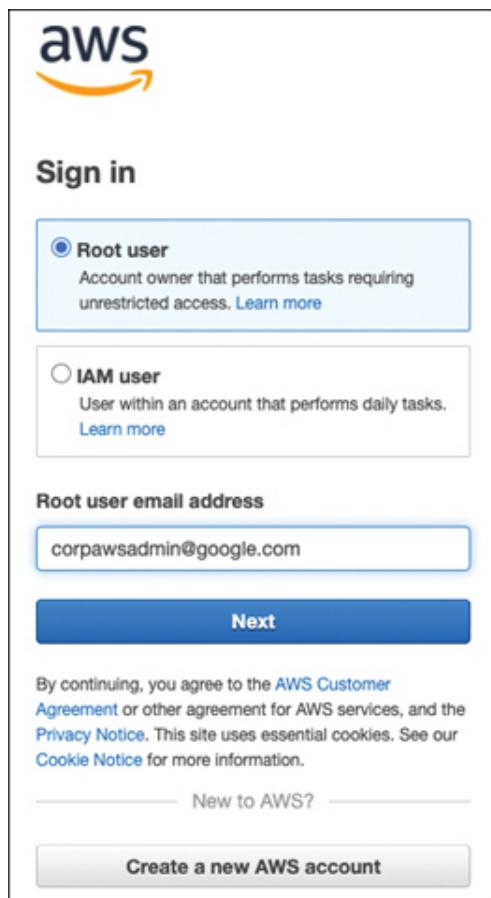
Every AWS account has an initial root user per AWS account created when each AWS account is first provisioned. The root user is the owner of the AWS account, and root credentials are the email address and password provided during the initial creation of each AWS account.

The first time you logged in to a new AWS account, you used the root account credentials to authenticate; after all, there weren't

any other administrator accounts available. Perhaps you're still using your root account credentials as your daily administrative account; however, doing so is not recommended, as the root user is not an IAM user controlled by IAM security. The root user has unrestricted access to all resources in the AWS account. Each root account has specific tasks that only the root account can perform. Think of the root account as a special administrator account that should only be used to perform specific tasks, such as billing, changing the AWS support plan, or reviewing tax invoices. The root user is not meant for daily administrative duties. If the root account is the only admin account available in your AWS account, you need to create several IAM users as administrators to properly safeguard your AWS account resources.

Here's a quick way to check if you're using an AWS account root account: What are the security credentials you use to log in to the AWS account? If you use an email address (see [Figure 3-8](#)), you are accessing this AWS account as the root user. Now think about how many other administrators could potentially be using the same root account login; each of these administrators could potentially delete everything in the associated AWS account when using the root user account logon. There's no way to disable root account actions because there are no IAM controls on the root account. And no controls can be added.

**Key  
Topic**



**Figure 3-8** Root User Logon

Why would AWS create an initial user account that has unlimited power? The first account in any operating system must have the most power; think of an Active Directory Domain Services enterprise administrator or the root user for the Linux operating system. As in any other operating system, the root

credentials need to be protected. AWS will alert you to create additional IAM users to protect your AWS account.

The following tasks can only be carried out in each AWS account when authenticated as the AWS root user:

- Modifying the root user account details, including changing the password of the root account
- Closing an AWS account
- Changing your AWS support plan from free to Developer, Small Business, or Enterprise
- Enabling billing for the account or changing your payment options or billing information
- Creating a CloudFront key pair
- Enabling MFA on an S3 bucket in your AWS account
- Requesting permission to perform a penetration test
- Restoring IAM user permissions that have been revoked

---

### Note

After you sign in for the first time using the root user for your AWS account, the best practice is to create an IAM user for administration duties, add the required administrative policies and privileges to your new IAM user account, and stop using the

root account unless it is necessary to carry out a root administrative task.

---

## The IAM User

An IAM user is an IAM security principal that can be used to access the following interfaces:

- Every IAM user with assigned username and password credentials can access AWS resources using the AWS Management Console.
- An IAM user with assigned username and password credentials and an active access key (access key ID and secret access key) is allowed both AWS Management Console access *and* programmatic access from the AWS CLI. Script or CLI commands will not execute until the IAM user account's access ID and secret access keys are validated.

There are two ways to identify an IAM user:

- The most common way is the name of the user account listed in the IAM dashboard. This username also shows up in each IAM group's list of associated IAM users.
- An Amazon Resource Name (ARN) uniquely identifies each IAM user across all AWS user accounts. Every resource that is

created at AWS also has a unique ARN. For example, if you create a resource policy to control access to an S3 bucket, you will need to specify the user's account ARN that can access the bucket in the following format: *arn:aws:iam::account ID:user/mark*.

## Creating an IAM User

The easiest way to start creating IAM users is to use the IAM dashboard and click Add Users, which opens the dialog box shown in [Figure 3-9](#).

Add user

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name\* Susan

[Add another user](#)

Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the console using an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

Select AWS credential type\*

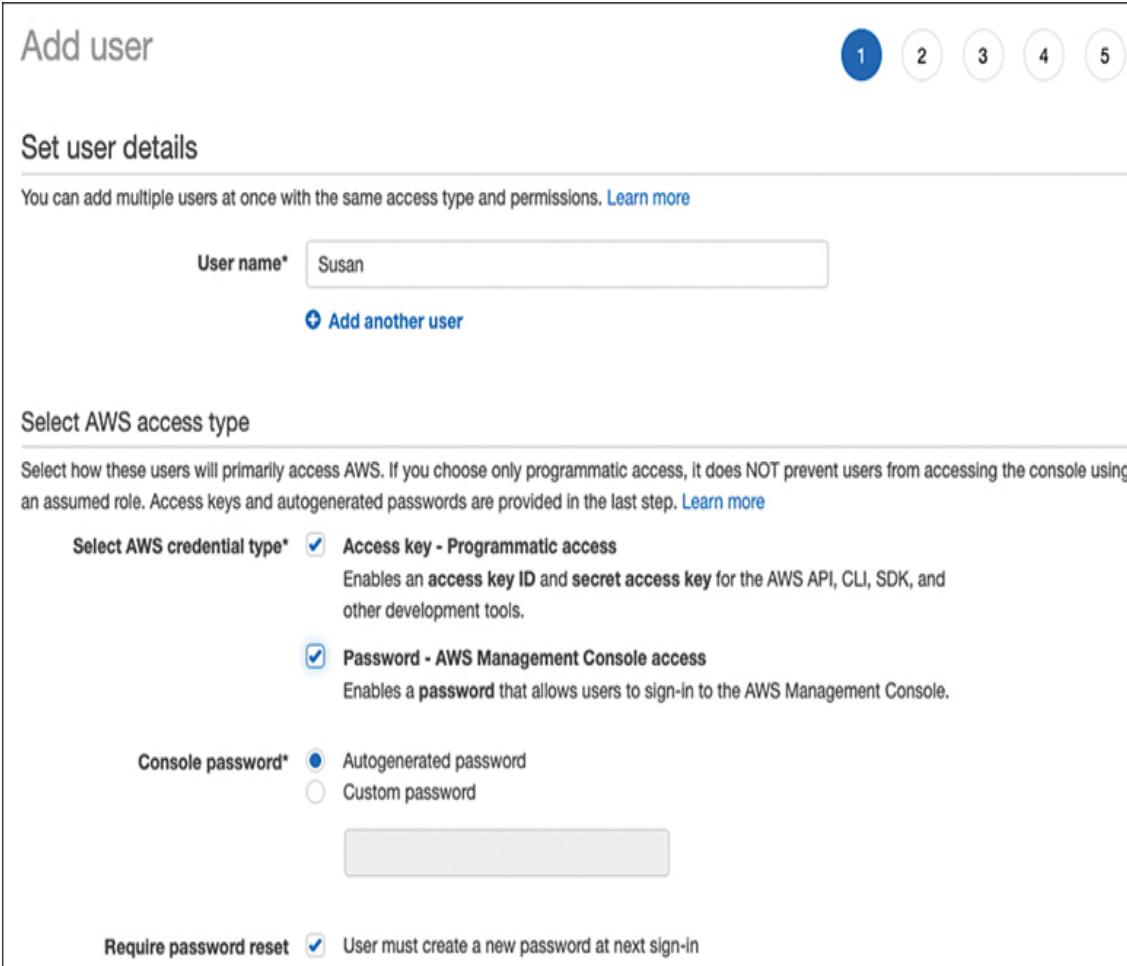
Access key - Programmatic access  
Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.

Password - AWS Management Console access  
Enables a **password** that allows users to sign-in to the AWS Management Console.

Console password\*

Autogenerated password  
 Custom password

Require password reset  User must create a new password at next sign-in



**Figure 3-9** Creating an IAM User

The first decision you must make is the type of access you want to allow your new IAM user to have:

- **Password – AWS Management Console access:** With this type of access, users enter a username and password to authenticate. If console access is all that is required, access

keys (an access key ID and secret access key) are not required.

- **Access key – Programmatic access:** This type of access also allows working from the command prompt using the AWS CLI or AWS SDK. Checking both boxes allows both types of access.
- 

### Note

If you're taking over an existing AWS environment, you might find that IAM users have access keys assigned to their accounts, but they don't actually carry out programmatic tasks. If this is the case, the current access keys can be deleted. In the future, if you decide that access keys are required, they can be added. It is a best practice to remove the root account access keys to make it impossible to run scripts and automation when logged in as a root user.

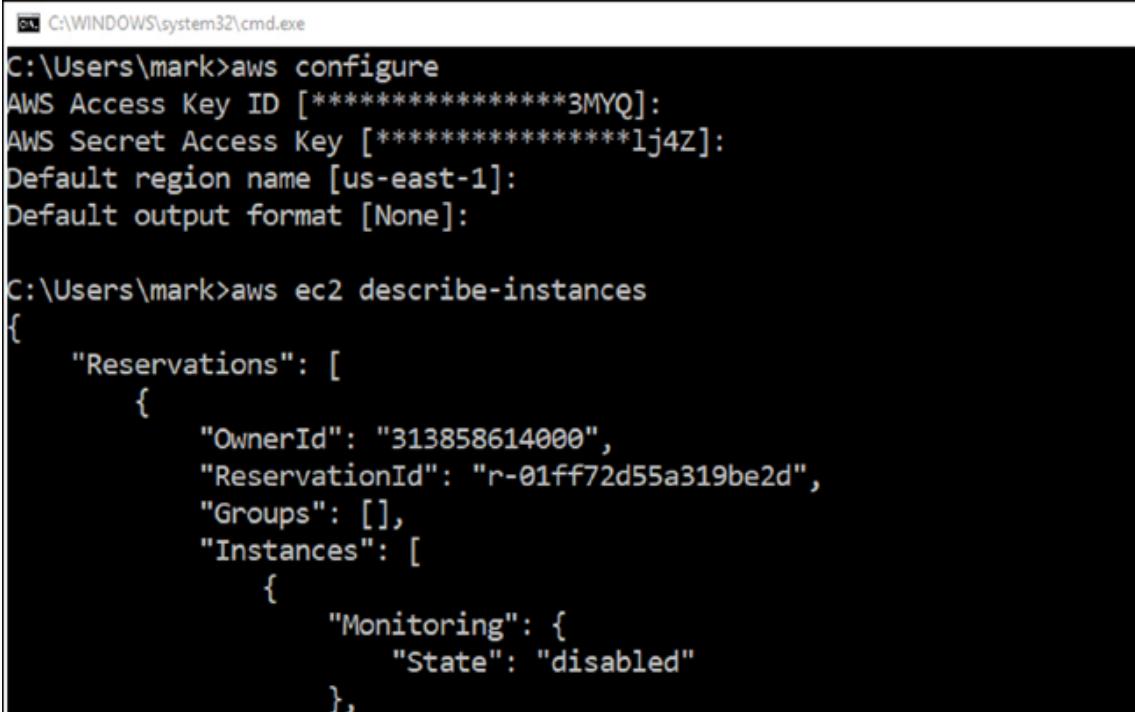
---

## IAM User Access Keys

Each user account can have two access keys: the access key ID and a secret access key. As discussed earlier in this chapter, access keys are also required when using the AWS CLI (see

[\*\*Figure 3-10\*\*](#), when running scripts, when running PowerShell scripts, or when calling AWS APIs directly or through an application.

**Key Topic**



```
C:\WINDOWS\system32\cmd.exe
C:\Users\mark>aws configure
AWS Access Key ID [*****3MYQ]:
AWS Secret Access Key [*****1j4Z]:
Default region name [us-east-1]:
Default output format [None]

C:\Users\mark>aws ec2 describe-instances
{
    "Reservations": [
        {
            "OwnerId": "313858614000",
            "ReservationId": "r-01ff72d55a319be2d",
            "Groups": [],
            "Instances": [
                {
                    "Monitoring": {
                        "State": "disabled"
                    },
                    "State": {
                        "Name": "pending"
                    }
                }
            ]
        }
    ]
}
```

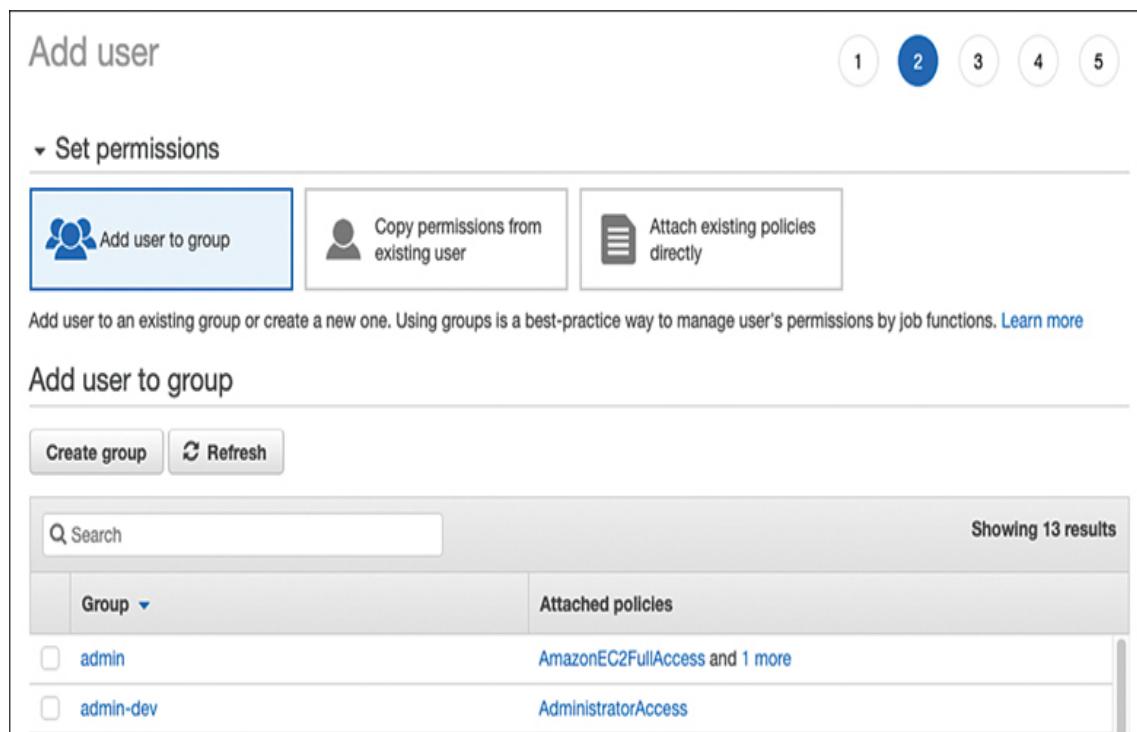
**Figure 3-10** Access Keys Required for CLI Operation

Once an IAM user account has been created successfully, you can download a copy of the access keys (access key ID and secret access key). This option is a one-shot deal: If you don't download a copy of the secret access key at the completion of

the user account creation process, you cannot view the assigned secret access key again. However, a new set of access keys for an already created IAM user (access ID and secret access key) can be requested.

There are three options available when creating a new IAM user using the IAM dashboard, as shown in [Figure 3-11](#):

- Add the user to an existing IAM group
- Copy permissions from existing IAM users to the user being created
- Attach existing policies directly to the new IAM user



**Figure 3-11** IAM User Account Creation Options

Creating a new IAM user without adding additional permissions or groups creates an extremely limited IAM user as there are no security policies assigned by default. One best practice to consider is to add the new IAM user to an existing IAM group that has the permissions needed for the IAM user account's required duties. Even if you are creating an IAM user for a specific AWS task that just this IAM user will carry out, you might want to think about adding this person to an IAM group if there's a possibility of multiple IAM users carrying out the same task in the future.

---

### Note

What each IAM user can and can't do at AWS is defined either by an explicit allow permission to carry out a task against AWS resources, or by explicit deny permissions that prohibit the user from being able to carry out a task. Note that an explicit deny for a specific task in any policy assigned to an IAM user account overrides any allow permissions defined in any other attached IAM policies.

---

## IAM Groups

An IAM group is a collection of IAM users. IAM groups are useful for delegating security policies to a specific group of IAM users. Attaching IAM groups to IAM user accounts makes assigning permissions much easier than having to modify each individual IAM user account. Each IAM user listed in an IAM group has their own authentication credentials and possible memberships in additional IAM groups. Each IAM group that IAM users are members of are assigned their IAM group permissions only after they have successfully authenticated to AWS. The characteristics of IAM groups are as follows:

- Each IAM group can contain multiple IAM users from the same AWS account.
- IAM users can belong to multiple IAM groups in the same AWS account.
- IAM groups can't be nested.
- IAM groups can only contain IAM users and not any additional IAM groups.
- There are initial quotas on the number of IAM groups you can have in each AWS account, and there is a quota that defines how many IAM groups an IAM user can be in. An IAM user can be a member of 10 IAM groups, and the maximum number of IAM users that can be created in a single AWS account is 5000.

## **Signing In as an IAM User**

After IAM users have been created, to make it easier for your IAM users to sign in to the IAM console, you can create a custom URL that contains your AWS account ID, as shown in [Figure 3-12](#).

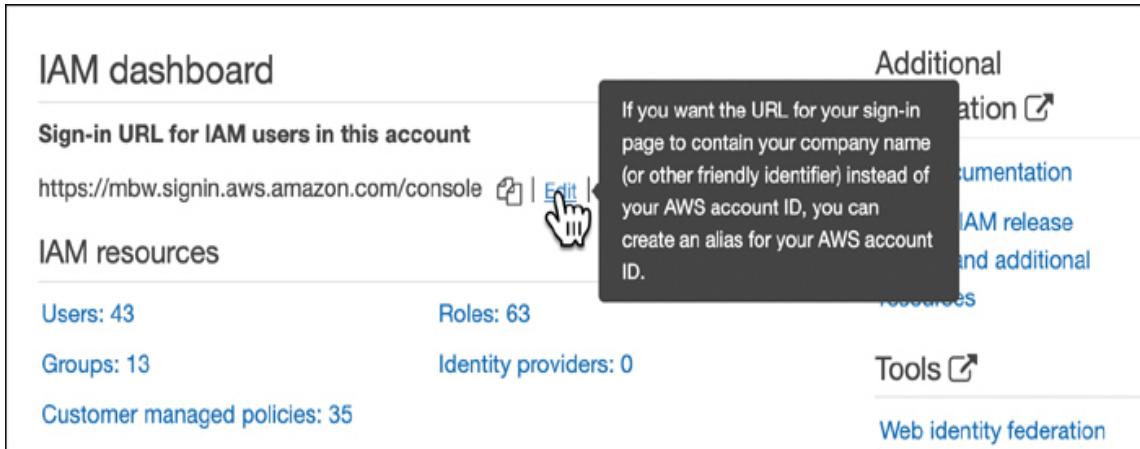
---

### **Note**

When creating and maintaining users, groups, and roles, you can manage IAM by using a third-party identity management product such as ForgeRock (<https://www.forgerock.com>), Okta (<https://www.okta.com>), or OneLogin (<https://www.onelogin.com>).

---





**Figure 3-12** Using Custom URL for IAM Users

## IAM Account Details

Each IAM user account displayed in the IAM console shows some useful information, including the IAM groups that the IAM user belongs to, the age of the access keys assigned to the IAM user account, the age of the current password, the last activity of the IAM account, and whether MFA has been enabled. Selecting an IAM user account in the console, you can see several additional account options, including the ARN of the account and information on the following tabs (see [Figure 3-13](#)):

- **Permissions:** This tab lists the applied permissions policies and the policy types.
- **Groups:** This tab lists the policies attached due to group membership.

- **Tags:** This tab lists key/value pairs (up to 50) that can be added for additional information.
- **Security Credentials:** This tab enables you to manage credential-related parameters such as the following:
  - The console password of the IAM user
  - The assigned MFA device (which can be a virtual or hardware device)
  - Signing certificates
  - Access keys
- **Access Advisor:** This tab lists the service permissions that have been granted to the IAM user and when the AWS services were last accessed within the calendar year.

The screenshot shows the AWS IAM User Summary page for a user named Susan. At the top, there's a breadcrumb navigation: 'Users > Susan'. Below it, the section title 'Summary' is displayed. Under the 'Summary' heading, there are three key details: 'User ARN' (arn:aws:iam::313858614000:user/Susan), 'Path' (/), and 'Creation time' (2022-06-17 11:56 EDT). A horizontal navigation bar below these includes tabs for 'Permissions', 'Groups', 'Tags (1)', 'Security credentials' (which is highlighted in red), and 'Access Advisor'. The main content area is divided into sections: 'Sign-in credentials' and 'Access keys'. The 'Sign-in credentials' section contains information about the console sign-in link, which is https://mbw.signin.aws.amazon.com/console. It also lists the status of the 'Console password' (Enabled, never signed in) and 'Assigned MFA device' (Not assigned), both with 'Manage' links. The 'Signing certificates' section indicates 'None'. The 'Access keys' section provides instructions for using access keys programmatically and cautions against sharing them. It includes a 'Create access key' button and a table showing one existing access key: 'Access key ID' (AKIAUSE3OMLYMV77IB6Z), 'Created' (2022-06-17 11:56 EDT), and 'Last used' (N/A).

Access key ID	Created	Last used
AKIAUSE3OMLYMV77IB6Z	2022-06-17 11:56 EDT	N/A

**Figure 3-13** User Account Summary Information

## Creating a Password Policy

***Password policies*** can be defined by selecting Account Settings in the IAM console. After password policies are defined, they control all IAM user accounts created in the AWS account. Password options include password complexity, password

expiration, password reuse, and whether IAM users can change their own passwords (see [Figure 3-14](#)).

Best practice is to review your corporate policy for passwords and consider whether more stringent rules need to be followed for working in the AWS cloud. If rules around password policy need to be tightened, the rules for the current on-premises password policy and the password policy defined in the AWS cloud should be analyzed and unified.

## Modify password policy

A password policy is a set of rules that define complexity requirements and mandatory rotation periods for your IAM users' passwords. [Learn more](#)

### Select your account password policy requirements:

Enforce minimum password length

6 characters

Require at least one uppercase letter from Latin alphabet (A-Z)

Require at least one lowercase letter from Latin alphabet (a-z)

Require at least one number

Require at least one non-alphanumeric character (! @ # \$ % ^ & \* () \_ + - = [ ] { } | ')

Enable password expiration

Expire passwords in 45 day(s)

Password expiration requires administrator reset

Allow users to change their own password

Prevent password reuse

Remember 2 password(s)

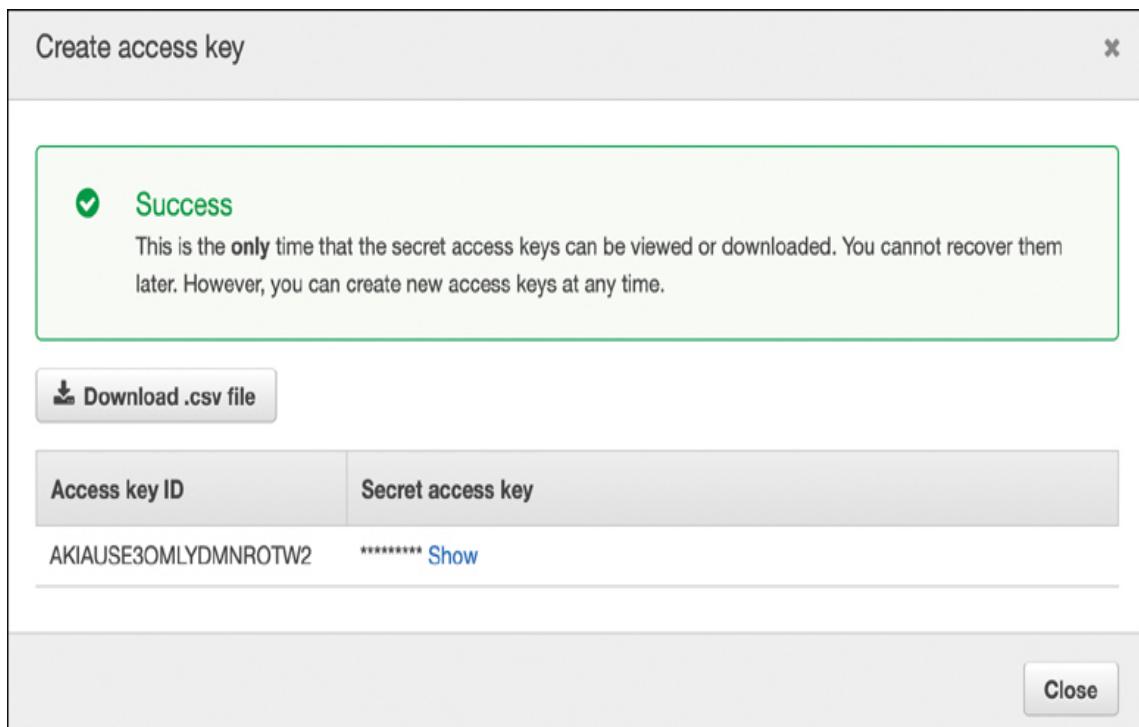
**Figure 3-14** Password Policy Options

## Rotating Access Keys

After an IAM user account has been created with access keys, the access keys are not changed unless they are manually rotated, or an automated process to perform the key rotation process is used such as a script or a custom Lambda function. Best practice is to rotate a user's access keys, preferably at the same time the IAM user password is changed, to maintain a higher level of security and avoid issues that can arise from compromised access keys. The access keys currently assigned to the IAM user can be viewed in the properties of the IAM user account on the Security Credentials tab. When a request is received to create a new access key, an associated secret access key is created along with the new access key ID, as shown in [Figure 3-15](#).

The important task of rotating access keys, shown in [Figure 3-16](#), should be assigned to a trusted IAM administrator account that will be carrying out the task of key rotation. Note that multiple **Get**, **Create**, **List**, **Update**, and **Delete** actions must be assigned to the selected IAM user in order to rotate access keys successfully.

## Key Topic



**Figure 3-15** Creating an Additional Access Key Manually

Actions Specify the actions allowed in IAM [?](#)

close

Switch to deny permissions [!](#)

Filter actions

Manual actions (add actions)

All IAM actions (iam:\*)

Access level

[Expand all](#) | [Collapse all](#)

GetAccountSummary [?](#)  ListGroupsForUser [?](#)  ListRoleTags [?](#)

GetLoginProfile [?](#)  ListInstanceProfiles [?](#)  ListSAMLProviders [?](#)

ListAccessKeys [?](#)  ListInstanceProfilesF... [?](#)  ListServerCertificates [?](#)

ListAccountAliases [?](#)  ListMFADevices [?](#)  ListServiceSpecificCr... [?](#)

ListAttachedGroupPo... [?](#)  ListOpenIDConnectPr... [?](#)  ListSigningCertificates [?](#)

ListAttachedRolePoli... [?](#)  ListPolicies [?](#)  ListSSHPublicKeys [?](#)

ListAttachedUserPoli... [?](#)  ListPoliciesGrantingS... [?](#)  ListUserPolicies [?](#)

ListEntitiesForPolicy [?](#)  ListPolicyVersions [?](#)  ListUsers [?](#)

ListGroupPolicies [?](#)  ListRolePolicies [?](#)  ListUserTags [?](#)

ListGroups [?](#)  ListRoles [?](#)  ListVirtualMFADevices [?](#)

Read (6 selected)

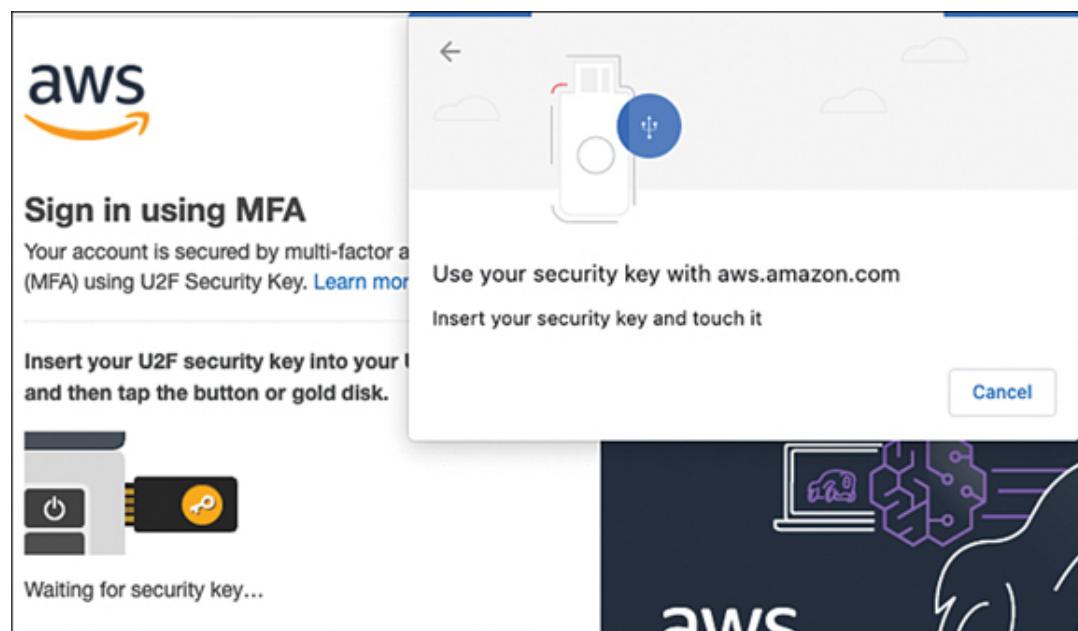
**Figure 3-16** Policy Actions for Rotating Access Keys

## Using Multi-Factor Authentication

**Key Topic**

Every AWS user account—including the root account and the IAM user account—supports MFA. With MFA enabled, during

the process of authenticating to AWS, a user must provide a security code in addition to the username and password credentials provided to access AWS resources, as shown in [Figure 3-17](#).



**Figure 3-17** Authenticating with MFA Enabled

There are several options available at AWS for deploying MFA:

- **Virtual MFA device:** A software app such as Google Authenticator or Authy, that typically is installed on the user's phone, can generate the six-digit code to be entered during authentication.
- **U2F security key:** A U2F security key is generated by a USB device that generates a security code when tapped. These

types of devices are approved by the Fast Identity Online (FIDO) Alliance. These keys are supported by many industry leaders, including Microsoft, Google, AWS, VMware, and Intel.

- **Hardware MFA device:** A hardware device such as a Thales SafeNet security appliance can also generate an MFA security code. Thales devices can provide end-to-end management of the entire encryption process.

## Creating IAM Policies

Identity and Access Management enables you to use or create a large number of security policies. Identity-based policies use the Identity and Access Service to apply security policies to an identified IAM user, group, or role.

---

### Note

The other type of security policy is called a resource-based policy. It is assigned to protect storage resources such as S3 buckets. Resource policies were available before the Identity and Access Management security service was introduced.

---

## IAM Policy Types

The actions for controlling AWS services with IAM policies are forever increasing as new features are added frequently to existing and new AWS cloud services. Make sure to check the documentation for each AWS service for the up-to-date choices. This section looks at the policy types that can be attached to IAM identities (users, groups, or roles).

### Identity-Based Policies

Identity-based policies are categorized as permission policies. Each identity-based policy contains permissions for specific actions an IAM user, group, or role can carry out. Policies can allow or deny access, and, optionally, indicate one or more mandatory conditions, must be met before access is allowed to the listed AWS cloud service or services defined in each policy.

There are three identity-based policy types:

- **Managed policies:** Managed policies, which are created and maintained by AWS, are read-only stand-alone identity-based policies that you can select and attach to IAM users, IAM groups, or roles created within each AWS account (see [Figure 3-18](#)). Listed are some concepts you need to understand when working with managed policies:

- Managed policies can be attached to and detached from any identity (that is, user, group, or role).
- A managed policy can be copied and saved as a custom policy.
- Managed policies cannot be deleted. When you detach a managed policy, it is removed from the selected identity, user, group, or role; however, the managed policy is still available in the library of managed AWS policies for reuse.
- Custom policies can be attached, detached, and deleted.



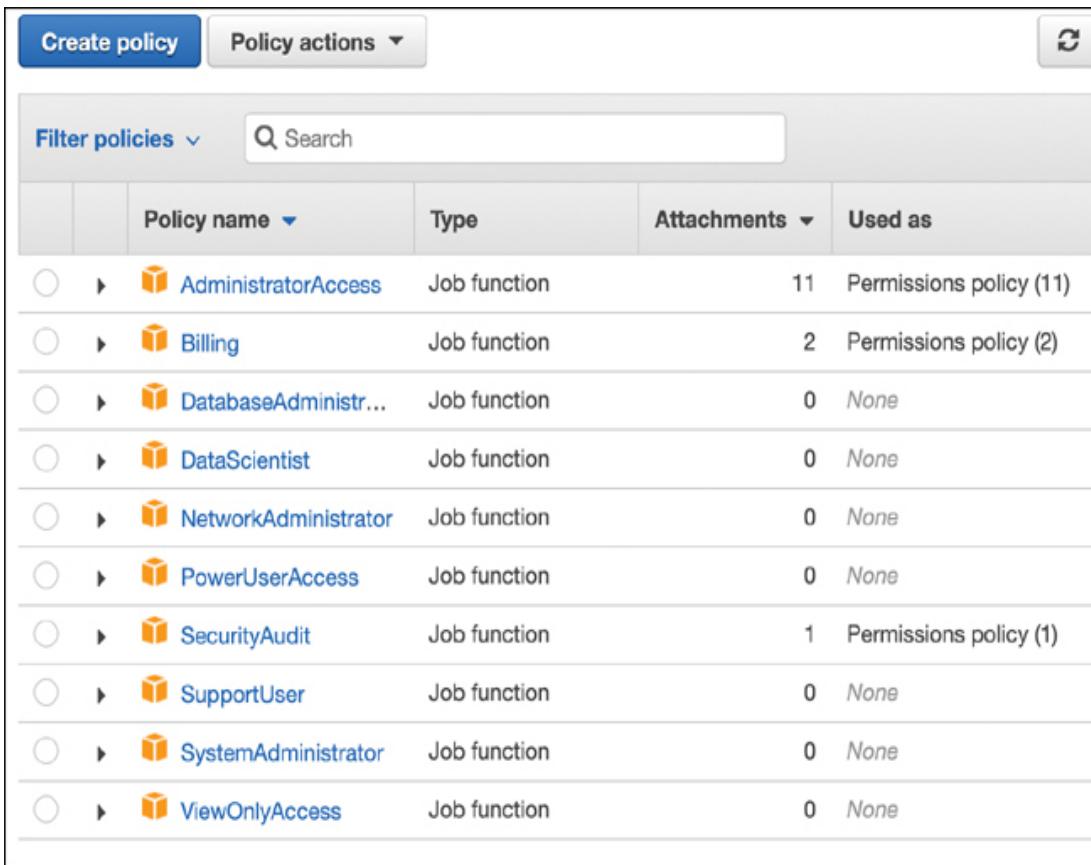
The screenshot shows the AWS IAM Policies page. At the top, there are buttons for "Create policy" and "Policy actions". Below that is a search bar and a "Reset filters" button. A sidebar on the left contains "Filter policies" and a "Search" field. The main area is a table with the following columns: Name, Attachments, and Used as.

	Attachments	Used as
Customer managed (35)	1 function	Permissions policy (1)
AWS managed (773)	0 managed	None
AWS managed - job function (10)	0 managed	None
Used for permissions (72)	1 managed	Permissions policy (1)
Used for boundary (0)	0 managed	None
Not used (746)	0 managed	None
AlexaForBusinessP...	AWS managed	0 None

**Figure 3-18** Managed Policies

- **Managed Policies for Job function:** Job function policies, which are also created and managed by AWS, are specialized managed policies based on generic job descriptions (see [Figure 3-19](#)). Job function policies might at first seem like an excellent idea. However, you need to be careful when assigning job function policies because a job function policy may assign more permissions than you need or wish to assign. For example, the SystemAdministrator job function policy allows the creation and maintenance of resources across many AWS services, including AWS CloudTrail, AWS

CloudWatch, AWS CodeCommit, AWS CodeDeploy, AWS Config, AWS Directory Service, Amazon EC2, AWS IAM, AWS Lambda, Amazon Relational Database Service (RDS), Amazon Route 53, AWS Trusted Advisor, and Amazon Virtual Private Cloud (VPC). However, a job function policy can be useful as a starting policy template that once imported as a custom policy enables you to make further modifications to suit your organization's needs. The job function policies that can be selected are Administrator, Billing, Database Administrator, Data Scientist, Developer Power User, Network Administrator, Security Auditor, Support User, System Administrator, and View-Only User.



The screenshot shows the AWS IAM console interface for managing job function policies. At the top, there are buttons for 'Create policy' and 'Policy actions'. Below that is a search bar with a placeholder 'Search' and a 'Filter policies' dropdown. The main area is a table listing ten job function policies:

	Policy name	Type	Attachments	Used as
<input type="radio"/>	AdministratorAccess	Job function	11	Permissions policy (11)
<input type="radio"/>	Billing	Job function	2	Permissions policy (2)
<input type="radio"/>	DatabaseAdministrat...	Job function	0	None
<input type="radio"/>	DataScientist	Job function	0	None
<input type="radio"/>	NetworkAdministrator	Job function	0	None
<input type="radio"/>	PowerUserAccess	Job function	0	None
<input type="radio"/>	SecurityAudit	Job function	1	Permissions policy (1)
<input type="radio"/>	SupportUser	Job function	0	None
<input type="radio"/>	SystemAdministrator	Job function	0	None
<input type="radio"/>	ViewOnlyAccess	Job function	0	None

**Figure 3-19 Job Function Policies**

- **Custom policies:** You can select any managed policy as a starting template, modify it for your requirements, saving it as a custom policy in your AWS account. You can also elect to start with a blank page when creating a custom policy document and create the entire policy from scratch using the IAM dashboard. Each custom policy created is managed and maintained by each organization.

## Resource-Based Policies

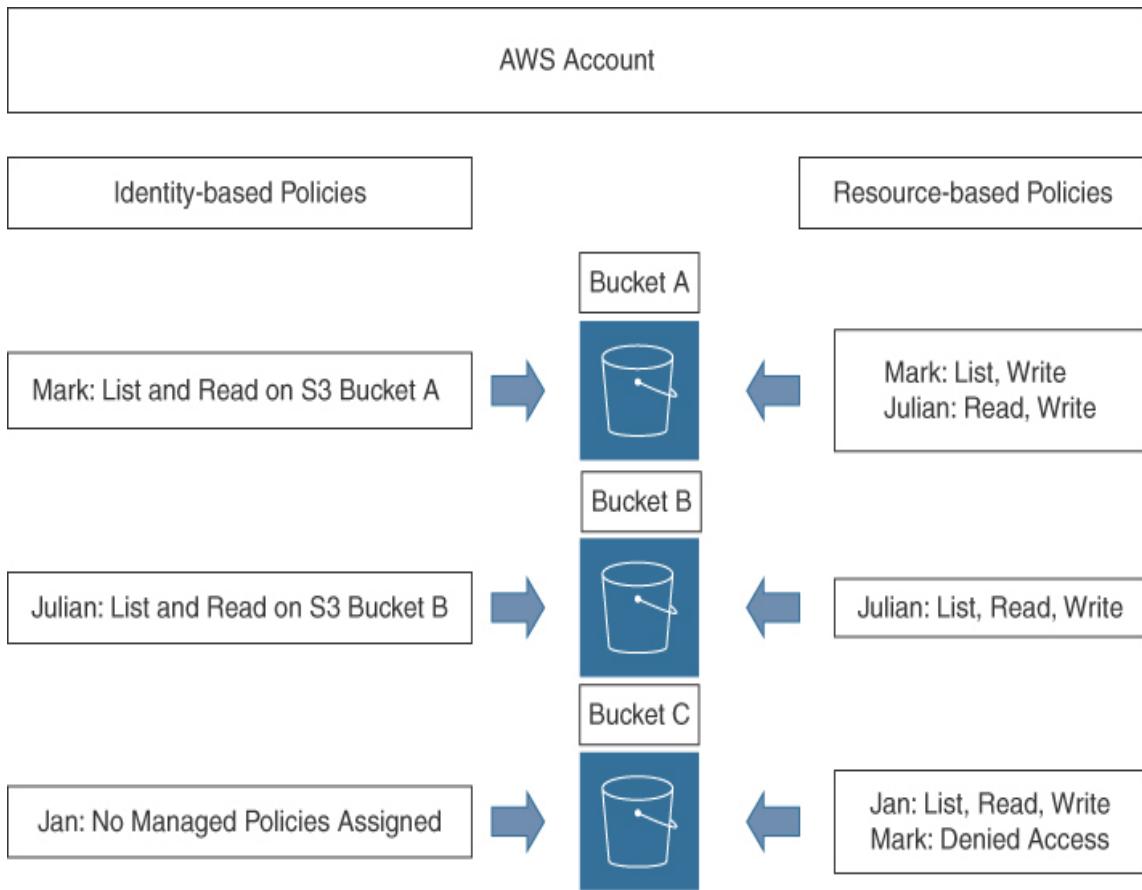
**Key  
Topic**

As previously discussed, identity-based policies are attached to an IAM user, group, or role defining what actions each attached identity is allowed, or not allowed to do. Resource-based policies are a little different in functionality because they are attached directly to AWS resources and are not created using the AWS Identity Access Management service. Resource-based policies are supported by several AWS storage services; the most common example is an Amazon S3 bucket, but there are other older AWS cloud services that support resource-based policies, including Amazon S3 Glacier vaults, Amazon Simple Notification Service (SNS), Amazon Simple Queue Service (SQS), and AWS Lambda functions. Because resource policies are attached directly to the AWS resource, each policy needs to define the access rules for the AWS resource and the IAM user, group, or AWS account that will access the resource. Resource-based policies are similar in functionality to IAM *inline policies* due to the direct attaching of the resource policy to the AWS resource; if a resource is deleted, the resource policy is unattached and discarded. Resource-based policies are always a custom creation; AWS does not create any managed resource-based policies. Inline policies are discussed later in this chapter.

An IAM user can be assigned both a managed IAM identity policy and a resource-based policy for accessing the same AWS resource (see [Figure 3-20](#)):

- IAM User Mark has an identity-based policy that allows him to list and read from S3 Bucket A.
- The resource—in this case, the S3 bucket—has an attached resource-based policy that identifies that Mark can list and write on S3 Bucket A.
- S3 Bucket C has an attached resource-based policy that denies access to Mark. IAM User Julian also has a combination of identity- and resource-based policies.
- IAM User Jan has no managed policies assigned.
- Jan has access to S3 Bucket C because she is specifically listed in the resource policy using her IAM User ARN.

**Key Topic**



**Figure 3-20** Identity and Resource Policies Working Together

---

### Note

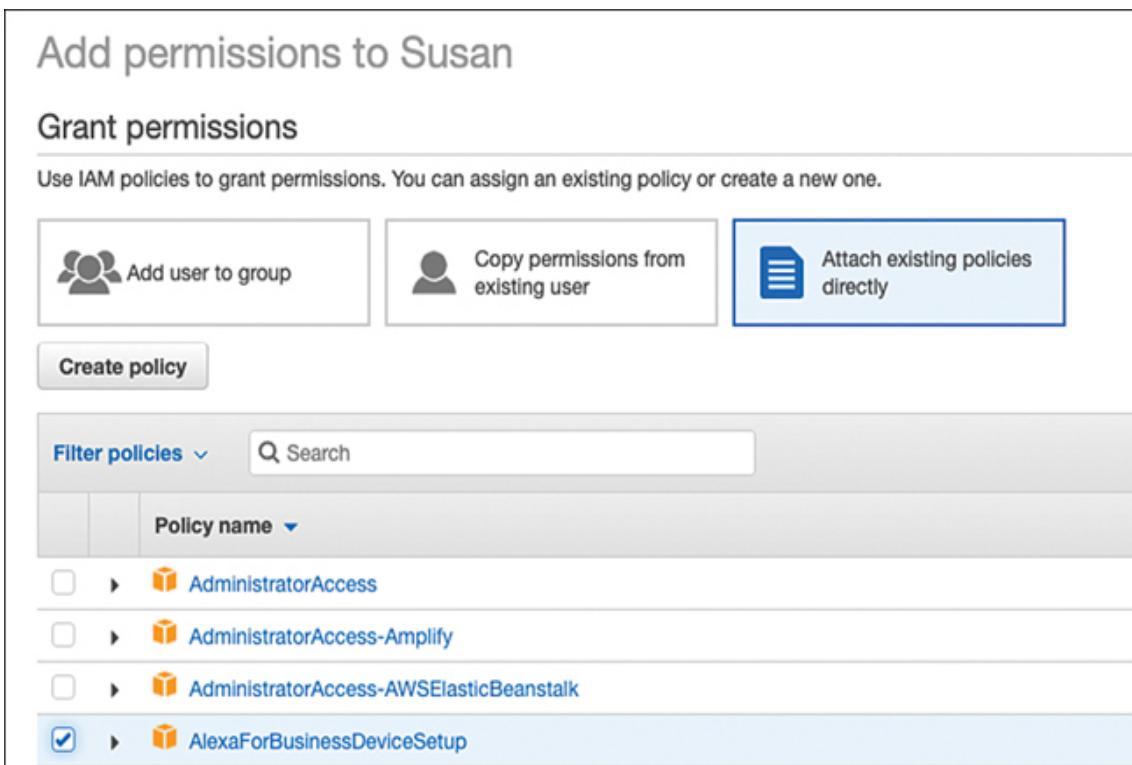
Amazon S3 bucket policies are resource policies.

---

## Inline Policies

Another method of attaching IAM policies is through the process of what is called an inline or directly attached policy, as shown in [Figure 3-21](#). An IAM policy that is attached inline

helps you maintain a strict one-to-one relationship between the attached policy and the entity the policy is attached to. When the entity is deleted, the attached policies are discarded. In comparison, using a managed policy allows you to apply the policy to multiple IAM users and groups.



**Figure 3-21** Attaching Existing Policies Directly

For example, a specific user with high security clearance within your organization has been assigned the task of managing AWS CloudHSM, a security service that uses single-tenant hardware security modules (HSMs) for storing your organization's

symmetric and asymmetric keys. You've decided to manage the security for this service by using inline policies that are attached to just one trusted administrator to ensure that only this person can carry out the specific tasks. Perhaps you have two security administrators, and you use inline policies to ensure that the policies are only assigned to these two individuals. You could use an IAM group but you don't want to make a mistake and accidentally add an additional IAM user to the existing group and weaken your security. If the administrator's IAM user accounts are deleted, the inline policies are discarded as well.

---

### Note

IAM roles (which are discussed later in this chapter in the section “[IAM Roles](#)”) are also attached directly to the IAM user or federated user.

---

## IAM Policy Creation

Each IAM policy is crafted in what is called a lightweight data interchange format, JavaScript Object Notation (JSON) format. You can create and view any existing IAM policies by using the IAM dashboard or by using the AWS CLI and using the commands **create-policy** or **list-policies**. If you are just starting

with AWS, it's probably best to start with the IAM dashboard, where you can easily view the IAM users and groups, policies, and roles. For crafting IAM policies using the AWS CLI, the AWS CLI command reference for Identity and Access management can be found here:

<https://awscli.amazonaws.com/v2/documentation/api/latest/reference/iam/index.html>.

Each IAM policy can define a single permission statement or multiple permission statements. When you create custom policies, it is important to keep them as simple as possible to start with; don't mix AWS resource types in a single policy just because you can. It's a good idea to separate custom policies by AWS resource type for easier deployment and troubleshooting. You can create IAM policies by using several methods:

- Create IAM policies by using the visual editor in the IAM console.
- Create IAM policies by using the JSON editor in the IAM console (see [Figure 3-22](#)).
- Create and add IAM policies by using standard copy and paste techniques to import policy settings in JSON format into your JSON editor.
- Create IAM policies by using a third-party IAM tool that has been installed and properly configured. After authenticating

to AWS using a recognized IAM user with valid access keys and appropriate administrative permissions, you can create IAM users, groups, and roles with third-party tools, such as OneLogin or Ping Identity, instead of using the IAM console.

The screenshot shows the AWS IAM console interface. At the top, it says "Policies > AdministratorAccess". Below that is a section titled "Summary". Under "Summary", there is a "Policy ARN" field containing "arn:aws:iam::aws:policy/AdministratorAccess" with a copy icon. There is also a "Description" field with the text "Provides full access to AWS services and resources.". Below the summary, there are four tabs: "Permissions" (which is selected), "Policy usage", "Policy versions", and "Access Advisor". Under the "Permissions" tab, there is a "Policy summary" button and a "JSON" button. The main area displays the JSON code for the policy:

```
1- {
2-     "Version": "2012-10-17",
3-     "Statement": [
4-         {
5-             "Effect": "Allow",
6-             "Action": "*",
7-             "Resource": "*"
8-         }
9-     ]
10 }
```

**Figure 3-22** The JSON Editor

**Key Topic**

## Policy Elements

Each IAM policy contains mandatory and optional elements that you need to understand and be familiar with:

- **Version:** (Mandatory) This element is the version of the policy language that the policy is using (see [Figure 3-23](#)). The latest policy language version is 2012-10-17; the date/time version number is added automatically to each policy document when you are manually creating a policy document using the IAM console. Add the latest version to all custom policies if they are created outside the IAM console to ensure that any new AWS features you are referencing in the custom policy are supported. If no version number is listed, the oldest IAM version number is used, which can potentially cause problems. For example, if you were using tags to determine access, or permission boundaries in a custom policy with no listed version number, these newer features would not work without the latest version number present at the top of the policy document.



**Figure 3-23** Version Information

- **Statement:** (Mandatory) Each IAM policy has at least a single statement; multiple statements are allowed in a policy. When beginning to craft custom policies, it might be cleanest or simplest to limit each policy document to a single policy statement.
- **Sid:** (Optional) This element is a unique ID statement for additional identification purposes.
- **Effect:** (Mandatory) The effect of any listed action is Allow or Deny.
- **Action:** (Mandatory) Each action lists the API call(s) that is allowed or denied.

- **Principal:** (Optional) The account, user, role, or federated user of the policy allows or denies access to a resource.
- **Resource:** (Mandatory) This element identifies the AWS resource that the actions in the statement apply to.
- **Condition:** (Optional) This element defines the absolute circumstances that must be met for the policy to be applied.

## Reading a Simple JSON Policy

You need to follow a number of syntax and grammatical rules when creating custom IAM policies. One missing brace {} or missed comma or colon can cause lots of pain when you're troubleshooting an IAM policy. These are the rules:

- Text values—that is, string values—are always encased in double quotes.
- A string value is followed by a colon.
- The data parts in a policy are defined as name/value pairs.
- The name and the value are separated with a colon (for example, “**EffectAllow- When data in a policy has multiple name/value pairs, the name/value pairs are separated using commas.
- Braces {} contain objects.
- Each object can hold multiple name/value pairs.**

- If square brackets are used, there are multiple name/value pairs, separated by commas.

Let's look at a simple IAM policy example in [Example 3-1](#) and explore its construction. Note that the numbers shown are just for identification purposes.

### Example 3-1 IAM Policy

[Click here to view code image](#)

```
1.{  
2. "Version": "2012-10-17",  
3. "Statement": {  
4. "Effect": "Allow",  
5. "Action": "s3>ListBucket",  
6. "Resource": "arn:aws:s3:::graphic_bucket"  
7. }  
8. }
```

Each policy starts with a left brace that defines the start of the policy statement block. A curly right brace denotes the end of the policy statement block. In [Example 3-1](#), line 1 and line 8 start and end the policy statement block.

Line 2 shows the current version of IAM policies; both **Version** and the version number are in quotation marks because the values within the quotes are string values. You can treat the version line in an IAM policy as a mandatory policy element. The version number is a name/value pair, and the name and the value are separated by a colon. Because there are multiple name/value pairs in this policy, there is a comma at the end of each line that contains a name/value pair (that is, lines 2, 4, and 5).

The first statement in the policy, line 3, is defined by “**Statement**” (note the quotation marks) followed by a colon (: ) and another inner left brace ( { ) that denotes the start of the statement block, which includes **Effect**, **Action**, and **Resource**:

- Line 4, “**Effect**” (note the quotation marks), followed by a colon (: ), is set to the value “**Allow**” (also in quotation marks). “**Effect**” can be set to either **Allow** or **Deny**.
- Line 5, “**Action**” in this policy, is set to allow the listing of an S3 bucket.
- Line 6, “**Resource**”, specifies that the resource being controlled by this policy is the S3 bucket **graphic\_bucket**.

The resource references the ARN—the unique Amazon name that is assigned to each resource at creation. Resource lines

in policies don't have commas because a resource is a name/resource listing, not a name/value pair.

Line 7, the right curly brace (}), indicates that the statement block is complete. The final right curly bracket that starts line 8 indicates that the policy statement block is complete.

## Policy Actions

When creating custom policies, you will typically have to provide several actions for the user to be able to carry out the required tasks. Take, for example, creating a policy for an administrator to be able to create, change, or remove their IAM user account password. The actions that need to be listed in the policy must include the following:

- **CreateLoginProfile:** The user needs to be able to create a login profile.
- **DeleteLoginProfile:** The user must be able to delete their login profile if they want to make changes.
- **GetLoginProfile:** The user has to be able to access the login profile.
- **UpdateLoginProfile:** After making changes, the user has to be able to update their login information.

For an IAM user to be able to perform administration tasks for a group of IAM users, the additional actions required include creating users, deleting users, listing users and groups, removing policies, and renaming or changing information. To be able to make changes to an AWS resource, you must be able to modify and delete. The statement in [Example 3-2](#) provides the details for this policy.



### Example 3-2 IAM Policy for Performing Administrative Tasks

[Click here to view code image](#)

```
"Statement": [
{
  "Sid": "AllowUsersToPerformUserActions",
  "Effect": "Allow",
  "Action": [
    "iam>ListPolicies",
    "iam>GetPolicy",
    "iam>UpdateUser",
    "iam>AttachUserPolicy",
    "iam>ListEntitiesForPolicy",
    "iam>DeleteUserPolicy",
  ]
}
```

```
"iam>DeleteUser",
"iam>ListUserPolicies",
"iam>CreateUser",
"iam>RemoveUserFromGroup",
"iam>AddUserToGroup",
"iam> GetUserPolicy",
"iam>ListGroupsForUser",

"iam>PutUserPolicy",
"iam>ListAttachedUserPolicies",
"iam>ListUsers",
"iam> GetUser",
"iam>DetachUserPolicy"
},
],
```



## Additional Policy Control Options

Several policy options give you great power in how you manage security options for IAM users and groups, including permission boundaries, service control policies, access control lists, and session policies.

## Permission Boundaries

Permission boundaries are used to mandate the security policies that can be applied to an IAM user or role.

You can apply a permission boundary policy for both the IAM user and IAM role within a single AWS account. Without a permission boundary being defined, the applied managed or custom policy defines the maximum permissions that are granted to each particular IAM user or role. Adding a permission boundary provides a level of control by filtering the permissions that can be applied. The IAM user or role can only carry out the actions that are allowed by *both* the assigned identity-based policy and the permission boundary policy. Therefore, the permission settings defined are controlled by a permission boundary policy that establishes the specific listing of permissions that can be applied.

For example, suppose you want administrator Mark to be able to manage Amazon S3 buckets and EC2 instances—and that's all. In this case, you need to create the custom policy shown in [Example 3-3](#), which defines the permissions boundary for Mark—namely, that he can fully administrate Amazon S3 buckets and EC2 instances.

### **Example 3-3** Mark's Permission Boundary

[Click here to view code image](#)

```
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": [  
                "s3:*",  
                "ec2:*"  
            ],  
            "Resource": "*"  
        }  
    ]  
}
```

Once a permission boundary has been added to Mark's IAM account, as shown in [Figure 3-24](#), the only two AWS services that Mark will have full administrative control over are Amazon S3 Buckets and EC2 instances. In the future, an IAM policy is added to Mark's account to enable him to work with AWS CloudTrail and create alerts and alarms. However, when Mark goes to carry out actions using AWS CloudTrail, if the current permission boundary has not been updated, listing that Mark can also use the CloudTrail service, then this action will

be denied. The permission boundary policy settings must match up with the IAM policy settings that are applied to Mark's IAM user account.

The screenshot shows the AWS IAM 'Users' interface with 'mark' selected. The 'Summary' tab is active, displaying basic user information: User ARN (arn:aws:iam::313858614000:user/mark), Path (/), and Creation time (2014-02-13 08:32 EDT). Below this, a navigation bar includes 'Permissions' (selected), 'Groups (2)', 'Tags', 'Security credentials', and 'Access Advisor'. The 'Permissions' section contains two expandable items: 'Permissions policies (5 policies applied)' and 'Permissions boundary (set)'. The 'Permissions boundary (set)' section is expanded, showing a note about controlling maximum permissions and links to 'Change boundary' and 'Remove boundary'. A single policy named 'EC2\_S3\_Control (Managed policy)' is listed under this section.

**Figure 3-24** Adding a Permission Boundary to a User Account

The permission boundary shown in [Figure 3-24](#) could be much more stringent; instead of listing full control, a permission boundary could mandate a specific listing of tasks that Mark could carry out for both Amazon S3 buckets and EC2 instances.

**Key Topic**

## AWS Organizations Service Control Policies

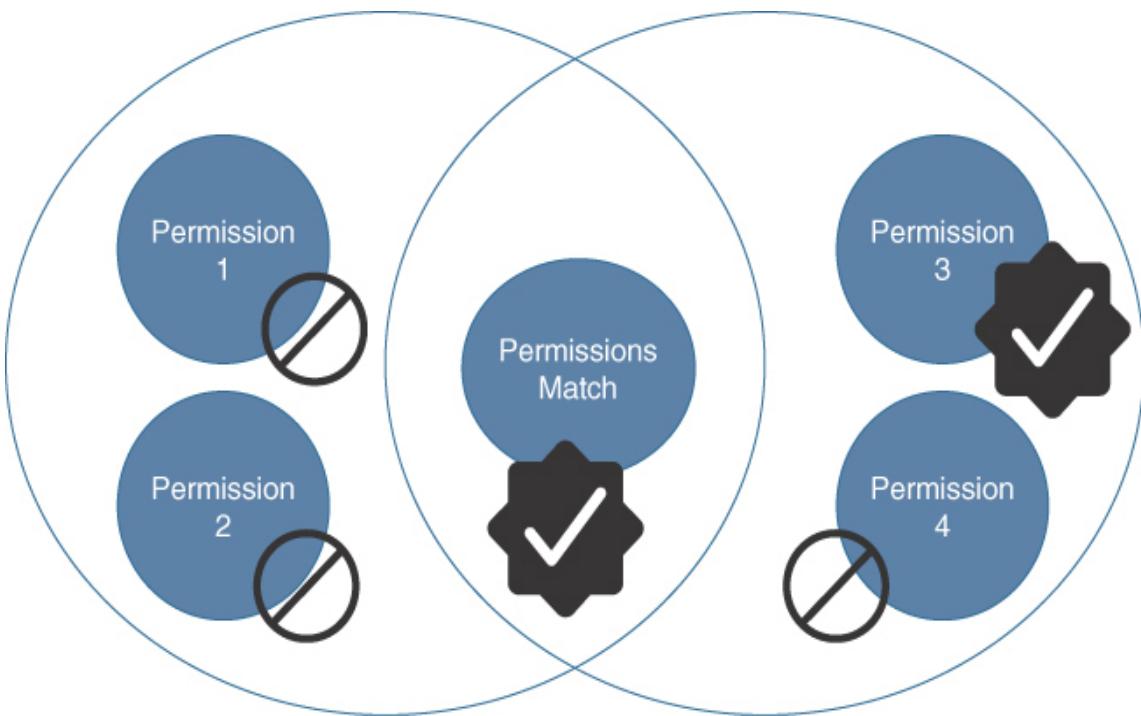
AWS Organizations enables organizations to manage security settings and services across AWS accounts that are grouped together in a tree formation. (More details on AWS Organizations are discussed later in this chapter.) One of the security features of AWS Organizations is a service control policy (SCP), which provides a permission boundary located at the root of the tree controlling all AWS account members, or to specific OUs containing AWS accounts in the AWS Organization tree. The SCP and the entity being controlled must have matching permissions for the desired permissions to be allowed (see [Figure 3-25](#)). Once an SCP has been enabled, permissions are allowed only if the IAM policy and the SCP list the identical permissions in both policies. The types of permission policies that can be controlled by an SCP are identity-based policies for IAM users, roles, the root user in any AWS account, and resource-based policies.

---

### Note

Service control policies do not affect service-linked roles that delegate the permissions assigned to each AWS service to carry out their assigned tasks.

---



**Figure 3-25** Effective Permissions Using a Service Control Policy

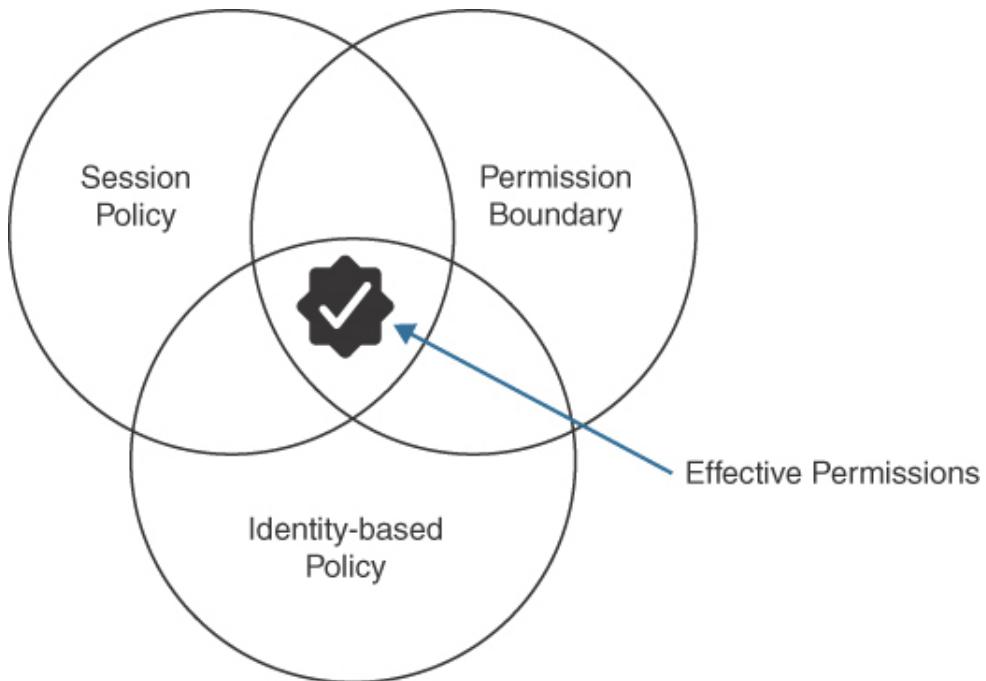
## Access Control Lists

Access control lists (ACLs) are present for defining simple permission controls on objects in Amazon S3 buckets for cross-account permission access only between separate AWS accounts. ACLs cannot be used to grant permissions to entities in the same AWS account. ACLs are only present because of backward compatibility; it's a much better idea to use IAM roles to control cross-account access. Amazon recommends that ACLs not be used for applying security to S3 bucket contents. Instead use the Amazon S3 Object Ownership setting **Bucket owner enforced** to disable all of the ACLs associated with a bucket.

When this bucket-level setting is applied, all of the objects in the bucket become owned by the AWS account that created the bucket and ACLs can no longer be used to grant access.

## Session Policies

Session policies are another version of a permission boundary to help limit what permissions can be assigned to federated users or IAM users assigned roles (see [Figure 3-26](#)). Developers can create session policies when IAM roles are used to access an application. When session policies are deployed, the effective permissions for the session are either the ones that are granted by the resource-based policy settings or the identity-based policy settings that match the session policy permission settings.

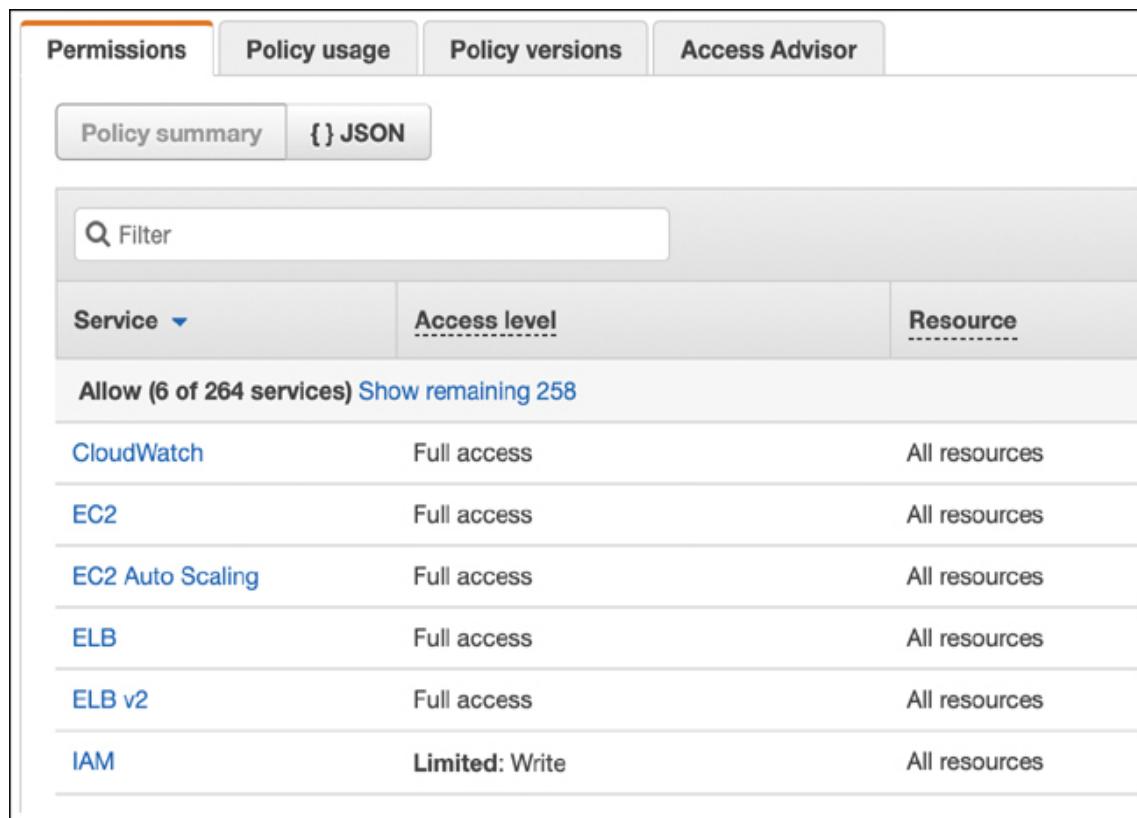


**Figure 3-26** Session Policies

## Reviewing Policy Permissions

For troubleshooting purposes, it may be necessary to review the assigned access levels, the required AWS resources, and any additional conditions that have been allowed or denied within each IAM policy. Thankfully, AWS provides these details in graphic *policy summary tables*, as shown in [Figure 3-27](#), which make it easier to troubleshoot or analyze what an IAM user, group, or role combined with a select IAM policy can do. There are policy summaries on both IAM users and roles for all attached policies. View a policy summary by selecting the individual policy; on its summary page click Policy Summary.

Information is displayed for the different types of policies: custom and AWS-managed policies and AWS-managed job function policies.



The screenshot shows the AWS IAM Policy Summary Tables interface. At the top, there are tabs for 'Permissions' (highlighted in orange), 'Policy usage', 'Policy versions', and 'Access Advisor'. Below the tabs are buttons for 'Policy summary' and '{ } JSON'. A search bar labeled 'Filter' is present. The main area contains three tables:

Service	Access level	Resource
CloudWatch	Full access	All resources
EC2	Full access	All resources
EC2 Auto Scaling	Full access	All resources
ELB	Full access	All resources
ELB v2	Full access	All resources
IAM	Limited: Write	All resources

Below the first table, a link 'Allow (6 of 264 services) Show remaining 258' is visible.

**Figure 3-27** Policy Summary Tables

Policy permissions information is contained in three tables:

- **Policy Summary (Services):** Information is grouped into explicit deny, allow, and uncategorized services when IAM can't figure out the service name due to a typo or when a custom third-party service is in use that has not been defined

properly. Recognized services are listed based on whether the policy allows or explicitly denies the use of the service.

- **Service Summary (Actions):** Information displayed includes a list of the actions and permissions (for example, list, read, write) that have been defined in the policy for a particular service.
- **Action Summary (Resources):** Information includes a list of resources and the conditions that control each action. Details include the resources, the region where the resources have been defined, and what IAM accounts the actions are associated with.

## IAM Policy Versions



After you've created an IAM policy, in the future you may want to make additions or deletions. Regardless of whether a policy is a custom policy that you have created or an AWS-managed policy, every time an IAM policy is updated, a new version of the policy is created.

AWS stores up to five versions of each IAM policy. To define the default version of an IAM policy to be used, after selecting the

policy, select the Policy versions tab, and from the displayed versions, select the version of the policy that you want to define as the current version to be used. From this point forward, the selected version of the policy becomes version enforced, as shown in [Figure 3-28](#). If you want to make changes later, you can change the current version of the policy to another version of the policy.

<b>Policy ARN</b>	arn:aws:iam::aws:policy/AmazonEC2FullAccess																			
<b>Description</b>	Provides full access to Amazon EC2 via the AWS Management Console.																			
<b>Permissions</b>	<b>Policy usage</b>	<b>Policy versions</b>																		
Each time you update a policy, you create a new version. You can have up to 5 versions. <a href="#">Learn more</a>																				
<table border="1"><thead><tr><th></th><th><b>Version</b></th><th><b>Creation time</b></th></tr></thead><tbody><tr><td>▶</td><td>Version 5 (Default)</td><td>2018-11-26 21:16 EST</td></tr><tr><td>▶</td><td>Version 4</td><td>2018-02-08 13:11 EST</td></tr><tr><td>▶</td><td>Version 3</td><td>2018-01-11 15:16 EST</td></tr><tr><td>▶</td><td>Version 2</td><td>2017-10-30 18:35 EST</td></tr><tr><td>▶</td><td>Version 1</td><td>2015-02-06 13:40 EST</td></tr></tbody></table>				<b>Version</b>	<b>Creation time</b>	▶	Version 5 (Default)	2018-11-26 21:16 EST	▶	Version 4	2018-02-08 13:11 EST	▶	Version 3	2018-01-11 15:16 EST	▶	Version 2	2017-10-30 18:35 EST	▶	Version 1	2015-02-06 13:40 EST
	<b>Version</b>	<b>Creation time</b>																		
▶	Version 5 (Default)	2018-11-26 21:16 EST																		
▶	Version 4	2018-02-08 13:11 EST																		
▶	Version 3	2018-01-11 15:16 EST																		
▶	Version 2	2017-10-30 18:35 EST																		
▶	Version 1	2015-02-06 13:40 EST																		

**Figure 3-28** Viewing Versions of IAM Policies

## Using Conditional Elements

Conditional elements of a JSON policy allow you to dictate optional parameters that must be met before the policy action is

approved. Conditional elements are global or service-specific, as shown in the examples in [Table 3-2](#). An organization could use the **aws:SourceIP** element, for example, to control the range of IP addresses from which administrators can log on to AWS.



**Table 3-2** Conditional Elements

Element	Description
<i>Global Elements</i>	
<b>aws:CurrentTime</b>	This element checks for date/time conditions.
<b>aws:SecureTransport</b>	The request must use Secure Sockets Layer (SSL/TLS).
<b>aws:UserAgent</b>	This element allows certain client

applications to make requests.

<b>aws:MultiFactorAuthPresent</b>	With this element, you can use the <b>BoolIfExists</b> operator to deny requests that do not include MFA.
-----------------------------------	---

<b>Bool</b>	The value of this element must be true.
-------------	---

<b>StringEquals</b>	The request must contain a specific value.
---------------------	--

### *Service-Specific Elements*

<b>aws:PrincipalOrgID</b>	With this element, the user must be a member of a specific AWS organization.
---------------------------	--

<b>aws:PrincipalTag/tag-key</b>	This element checks
---------------------------------	---------------------

for specific tags.

**aws:RequestTag/tag-key**

This element checks for a tag and a specific value.

**aws:PrincipalType**

This element checks for a specific user or role.

**aws:SourceVpce**

This element restricts access to a specific endpoint.

**aws:RequestedRegion**

This element allows you to control the regions to which API calls can be made.

**aws:SourceIp**

This element specifies an IPv4 or IPv6 address or range of addresses.

**aws:userid**

This element checks  
the user's ID.

## Using Tags with IAM Identities

Most AWS resources allow you to define a number of tags for the resource you are creating or using. You can add custom attributes using tags to both the IAM user and roles; for example, you can define a tag for an EC2 instance with the key **location** and the tag value **Toronto**.

Once you have tagged your resources, tags can be used to control IAM users and roles and their access to AWS resources. Tags can be added as a conditional element of each policy, mandating what tags need to be attached to the resource before the request is allowed. The following logic can be controlled using conditional tags:

- **Resources:** Tags can be used for IAM users and roles to determine whether access is allowed or denied to the requested resource based on the attached tags.
- **Principals:** Tags with Boolean logic can be used to control what the IAM user is allowed to do.

In [Example 3-4](#), administrators can only delete users who have the **ResourceTag** set to **temp\_user=can\_terminate** tag and **PrincipalTag** attached to **useradmin=true**. The tags in the example have been bolded for ease of reading.

### Example 3-4 Using Tags to Control Deletions

[Click here to view code image](#)

```
{  
    "Version": "2012-10-17",  
    "Statement": [{  
        "Effect": "Allow",  
        "Action": "iam:DeleteUser",  
        "Resource": "*",  
        "Condition": {"StringLike": {"iam:ResourceTag/temp_user":  
            "can_terminate"} }  
    }]  
}  
{  
    "Version": "2012-10-17",  
    "Statement": [  
        {  
            "Effect": "Allow",  
            "Action": "iam:* ",  
            "Resource": "*",  
            "Condition": {"StringEquals": {"aws:PrincipalTag/useradmin":  
                "true"} }  
        }]  
}
```

```
}
```

```
]
```

```
}
```

## IAM Roles



An ***IAM role*** is an IAM identity with specific permissions that define what the identity can and can't do at AWS. IAM roles provide temporary access to AWS resources once a role is associated with the following identities:

- An IAM user in the same AWS account as the role
- An IAM user in a different AWS account than the role
- An AWS web service such as Amazon EC2
- An external user authenticated by an external identity provider (IdP) service compatible with SAML 2.0 or OpenID Connect

When IAM roles are assumed by an identity, there is an additional linked policy called a *trust policy*. The use of an IAM role establishes a trust relationship between your *trusting*

account and other AWS *trusted* accounts. The trusting account owns the AWS resource to be accessed. The trusted account contains the IAM identity that needs access to the resource. The trust policy and the security policy are assigned to the identity who will assume the role, as shown in [Example 3-5](#). Roles do not have attached credentials. Temporary authentication credentials and a session token are assigned to an IAM user or federated user only after verification that the identity can assume the role. Trust policies are created for roles as follows:

- When a role is set up using the IAM console, the trust policy document is created and applied automatically.
- When a role is assigned to a user in the same AWS account, no trust policy is required, as the IAM user is already known to the AWS account.
- When a role is assigned to an IAM user residing in another AWS account, a trust policy must be assigned to the IAM user to be able to gain access.
- When the AWS CLI is used to create a role, both the trust policy and the permissions policy must be created.

### **Example 3-5 IAM Role Trust Policy**

[Click here to view code image](#)

```
{  
    "Version": "2012-10-17",  
    "Statement": {  
        "Effect": "Allow",  
        "Principal": {"AWS": "arn:iam::123456789:root"},  
        "Action": "sts:AssumeRole",  
    }  
}
```

## When to Use IAM Roles



IAM roles are used for these authentication scenarios:

- Access to AWS resources using service-linked roles
- EC2 instances hosting applications needing access to AWS resources
- Third-party access required to AWS Accounts resources
- Web identity federation authentication by an external identity provider requiring access to AWS resources
- SAML 2.0 federation authentication requiring access to AWS resources

- Cross-account access—AWS account identities requiring access to resources in another AWS account

The following sections describe these scenarios.

## AWS Services Perform Actions on Your Behalf

*Service-linked roles* assign the required permissions that allow each AWS service to carry out its job. AWS Config, Amazon Inspector, Amazon CloudWatch logs, and Amazon Elastic File System (EFS) are examples of AWS services using service-linked roles with the required permissions attached and temporary credentials granting access to carry out the requested tasks as required.

## EC2 Instances Hosting Applications Need Access to AWS Resources

AWS roles are useful for EC2 instances hosting applications that need access to AWS resources. For a workload to function properly, it needs valid AWS credentials to make its API requests to AWS resources. You could (but this a bad idea!) store a set of IAM users' credentials on the local hard disk of the application server or web server and allow the application to use those credentials.

Instead, implement the recommended best practice and create an IAM role that provides the required permission for the application hosted on the EC2 instance. The addition of a role to an EC2 instance creates an *instance profile* that is attached to the instance either during creation, as shown in [Figure 3-29](#), or after creation.

**Key Topic**

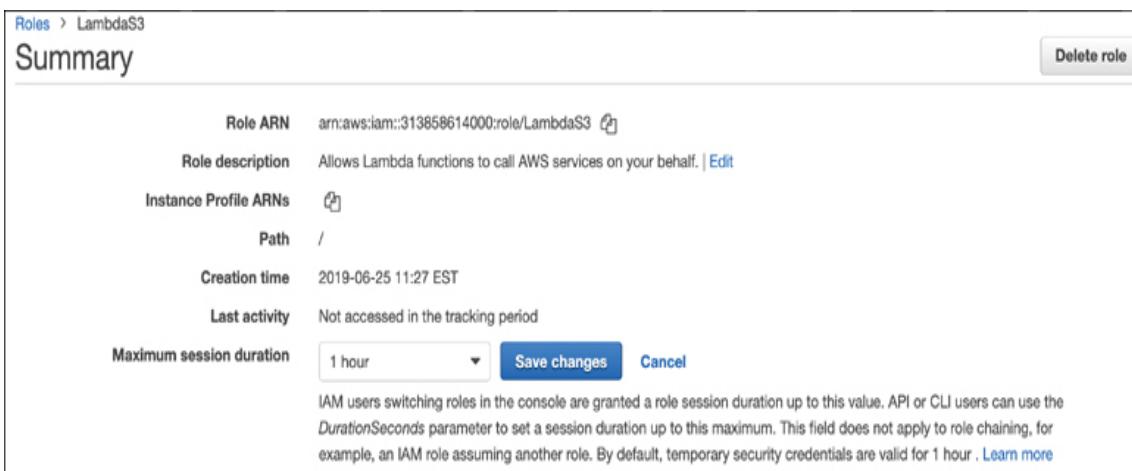
The screenshot shows the configuration options for creating a new Amazon EC2 instance. The 'IAM role' dropdown menu is open, displaying the selected role 's3\_access'. Other visible options include 'Network' (vpc-6d30d915 | Dev VPC), 'Subnet' (subnet-265f5f7c | Private Subnet 2 | us-east-1b), 'Auto-assign Public IP' (Use subnet setting (Disable)), 'Placement group' (checkbox for adding to placement group), 'Capacity Reservation' (Open dropdown), 'Domain join directory' (No directory), and 'Create new IAM role' (button).

**Figure 3-29** Attaching an IAM Role to an EC2 Instance

When the IAM role is used, temporary credentials are supplied, and the application can access the required AWS resources. Each EC2 instance can have a single role assigned; however, the single role can be assigned to multiple instances. Any future

changes made to the role are propagated to all instances that are currently using that role.

Using IAM roles means that you don't have to manage credentials. Instead, the AWS Security Token Service (STS) handles the authentication and authorization management. Each role assigned to an EC2 instance contains a permissions policy that lists the permissions to be used, plus a trust policy that allows the EC2 instance to be able to assume the assigned role and access the required AWS service. Each approved role is allowed access for a defined period of time; 1 hour is the default, as shown in [Figure 3-30](#). The temporary credentials are stored in the memory of the running instance and are part of the instance's metadata store under iam/security-credentials/role-name.



**Figure 3-30** Changing the Validity Time Frame for Temporary Credentials

Using temporary security credentials for an EC2 instance provides an additional advantage: The security credentials are automatically rotated just before their temporary session expires, ensuring that a valid set of credentials is always available for the application. IAM roles that control web/application server access to AWS cloud services is a concept that the AWS Certified Solutions Architect – Associate (SAA-C03) exam will expect you to know and understand.

## **Access to AWS Accounts by Third Parties**

Roles can be used to delegate access to third parties that require access to an organization's AWS resources. Perhaps the third party is managing some of your AWS resources. Granting access with a role and temporary security credentials allows you to grant access without sharing existing IAM security credentials. The role for the third party requires the following information:

- The third party's AWS account ID. The permissions policy specifies that identities from this AWS account number can assume the role.
- A secret identifier specified in the trust policy. The secret identifier is known to both the secure token service (AWS STS) and the third party.

- The permissions required by the third party to carry out their tasks.

## Web Identity Federation



Mobile applications can be designed to request temporary AWS security credentials using a process called *web identity federation*. Temporary credentials can map to a role with the required permissions to allow the mobile application to carry out its required tasks. Amazon Cognito is designed for scalable and secure mobile web-based federation to provide authentication for hundreds of thousands of users using social media providers such as Google, Facebook, Amazon, or any third-party identity provider that supports the OpenID Connect protocol. Amazon Cognito also provides support for enterprise federation using Microsoft Active Directory and any external IdP that supports SAML 2.0. Amazon Cognito uses user pools and federated identities to manage sign-up and authentication to mobile applications:

- **Amazon Cognito user pools:** Cognito enables you to create user pools of email addresses or phone numbers that can be

linked to the desired application along with the type of authentication needed: through the user pool or by federating through a third-party IdP.

- **Amazon Cognito federated identities:** Cognito manages multiple IdPs—both identity federation and web-based federation options that mobile applications use for authentication and controlling access to your backend AWS resources and APIs, ensuring users get only the requested access to AWS services such as Amazon S3, Amazon DynamoDB, Amazon API Gateway, and AWS Lambda (see [Figure 3-31](#)).



The screenshot shows the 'Create a user pool' page in the AWS Cognito console. On the left, a sidebar lists various configuration options: Name, Attributes (which is selected and highlighted in yellow), Policies, MFA and verifications, Message customizations, Tags, Devices, App clients, Triggers, and Review. The main content area is titled 'How do you want your end users to sign in?'. It explains that users can sign in with an email address, phone number, username or preferred username password. Below this, there are two sections: 'Username' and 'Email address or phone number'. Under 'Username', three checkboxes are available: 'Also allow sign in with verified email address', 'Also allow sign in with verified phone number', and 'Also allow sign in with preferred username (a username that your users can change)'. The first checkbox is checked. Under 'Email address or phone number', three radio buttons are available: 'Allow email addresses', 'Allow phone numbers', and 'Allow both email addresses and phone numbers (users can choose one)'. The third radio button is selected.

**Figure 3-31** Using Cognito for Mobile User Authentication

## SAML 2.0 Federation

The changes in authentication over the past 20 years have led to a number of options for single sign-on, including Cognito, AWS STS, Web Identity Federation, SAML 2.0, and Open ID Connect. Many companies use Active Directory Domain Services, which has supported SAML for many years. SAML is supported by all public cloud providers in order to support most major corporations' ability to authenticate to the cloud using SSO.

Before the rise of mobile phones, corporate computer/user accounts were linked to applications hosted in the cloud. Mobile applications on devices are now commonplace requiring a

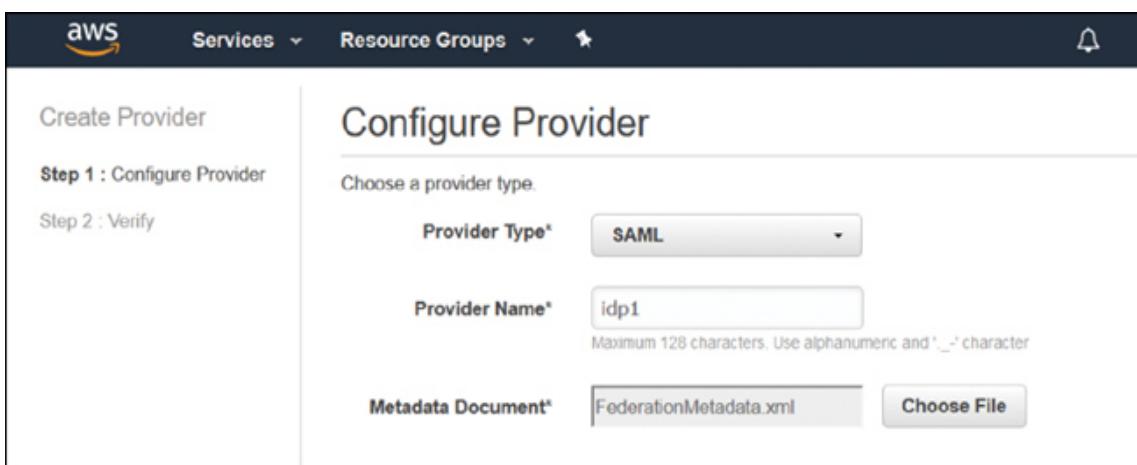
unique type of authentication linking phones and devices running an application hosted by the cloud provider.

If an organization's end users already authenticate to a corporate network using a security service such as AD DS, you don't have to create separate IAM users for access to AWS services and resources. Instead, your users' corporate Active Directory user identities can be *federated* and synchronized to AWS with access to AWS resources using IAM roles. If your corporate network is compatible with SAML 2.0, it can be configured to provide an SSO process for gaining access to the AWS Management Console or other AWS services as required.

AWS provides several services to handle the different levels of federation used today. Amazon Cognito allows you to manage the variety of authentication providers in the industry, including Facebook, Google, Twitter, OpenID, and SAML, and even custom authentication providers that can be created from scratch. The odds are that you will use several of these prebuilt third-party authentication providers for controlling authentication and access to applications hosted at AWS. AD DS deployments with Active Directory Federated Services installed can take advantage of AWS Directory Service to build a trust relationship between your corporate Active Directory network, your corporate users, and resources hosted in an AWS account.

Here are big-picture steps for linking your on-premises Active Directory environment with AWS. Registration of the organization identity provider with AWS is necessary before you can create IAM roles that define the tasks that your corporate users can carry out at AWS.

**Step 1.** Register your organization's identity provider, such as Active Directory, with AWS. To do so, you must create and provide a metadata XML file, as shown in [Figure 3-32](#), that lists your IdP and authentication keys used by AWS to validate the authentication requests from your organization.

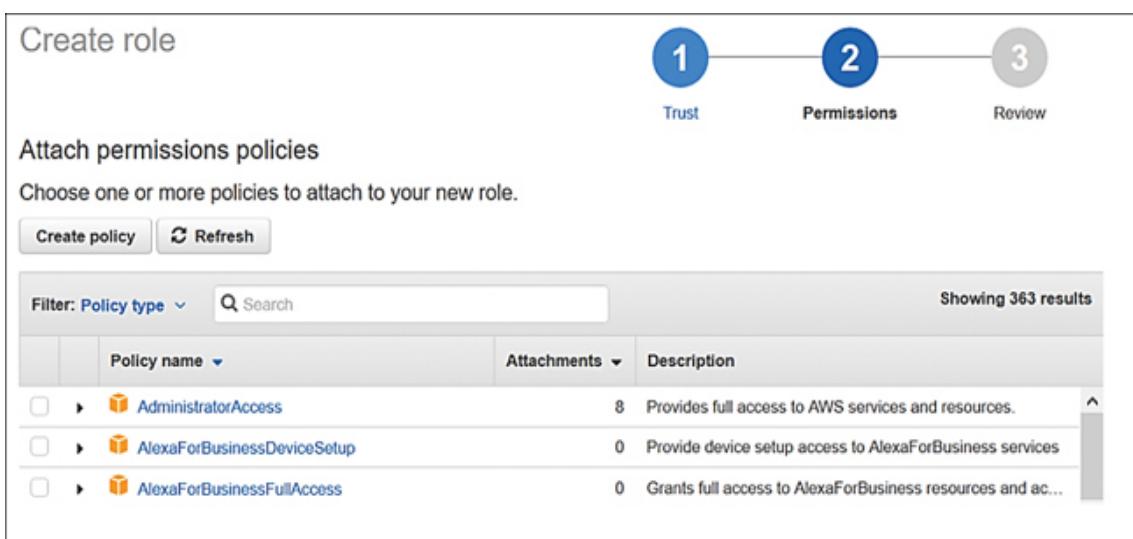


**Figure 3-32** Adding a Metadata XML

**Step 2.** Create IAM roles that provide access to the AWS resources. For the trust policy of the role, list your IdP as the

principal. This ensures that users from your organization will be allowed to access AWS resources.

**Step 3.** Define which users or groups to map to the IAM roles, as shown in [Figure 3-33](#), to provide access to the required AWS resources.



**Figure 3-33** Selecting Policies for an IAM Role

## Cross-Account Access

To allow access to resources in your AWS account from users in other AWS accounts rather than create an IAM user account within multiple AWS accounts for access, you can instead provide temporary *cross-account access* by using an IAM role. For example, a developer's IAM account located in the dev AWS account needs access to the S3 bucket **corpdocs** in the

production AWS account. User identities in the dev AWS account use IAM roles to assume access to AWS resources in the production AWS account, using defined IAM roles and policies that authenticate using AWS STS. The following steps allow access to specific AWS services hosted in the production account from the dev AWS account:

**Step 1.** Create an IAM policy called **access-s3** in the production account that controls access to the S3 resource. The policy created is a custom policy that allows access to a specific S3 resource, as shown here:

[Click here to view code image](#)

```
Statement": [
  {
    "Effect": "Allow",
    "Action": "s3>ListAllMyBuckets",
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3>ListBucket",
      "s3>GetBucketLocation"
    ],
    "Resource": "arn:aws:s3:::corpdocs"
```

```
 },
{
  "Effect": " Allow",
  "Action": [
    "s3:GetObject",
    "s3:PutObject",
    "s3:DeleteObject"
  ],
  "Resource": "arn:aws:s3:::corpdocs/*"
}
]
```

**Step 2.** Create an IAM role called **get-access**, which is assigned to the developer's IAM account that is linked to the IAM role policy **access-s3**.

**Step 3.** Get the ARN of the **get-access** role. The ARN is required to populate the custom IAM policy that allows the developer's IAM group to successfully switch accounts and access the **get-access** role.

**Step 4.** Grant access to the role in the developer's IAM user account by creating a custom policy that allows the developer to access the **get-access** role, as shown here:

[Click here to view code image](#)

```
{  
  "Version": "2012-10-17",  
  "Statement": {  
    "Effect": "Allow",  
    "Action": "sts:AssumeRole",  
    "Resource": "arn:aws:iam::::PRODUCTION-AWS-ACCT-ID:rol  
  }  
}
```

**Step 5.** The developer can now switch roles by using the AWS Management Console and clicking Switch Role below the username, as shown in [Figure 3-34](#), to gain access to the desired AWS resource.

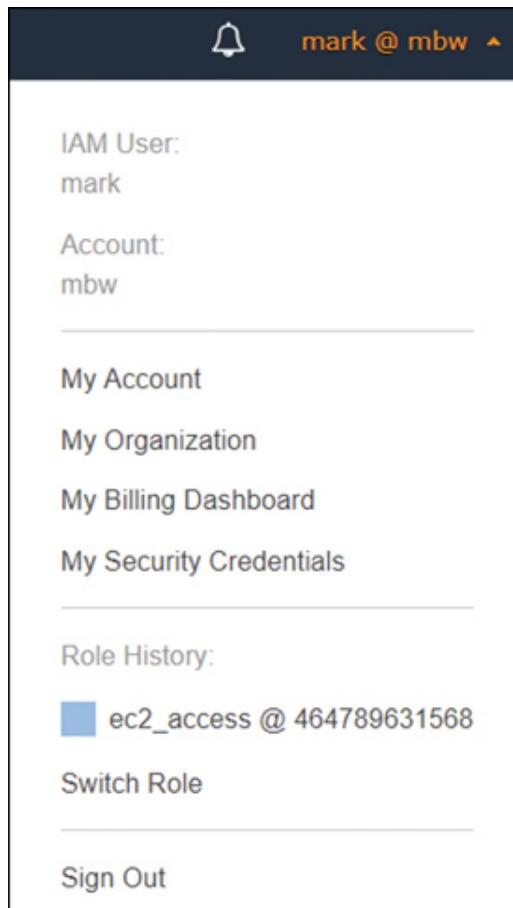
---

### Note

All Amazon services except for AWS RoboMaker, Amazon QuickSight, AWS Amplify, and Amazon Rekognition allow the use of roles. The action in the policy **AssumeRole** triggers communication with STS for verification.

---





**Figure 3-34** Using the Switch Role Option for Cross-Account Access

**Key Topic**

## AWS Security Token Service

External authentication using identity federation is possible at AWS, including SSO federation with SAML 2.0, web-based federation (via Amazon, Google, or Facebook), and federation using OpenID Connect. To support the various types of identity

federation, running in the background at AWS is a global security service that provides temporary credentials upon request for external and internal access to AWS services using an attached IAM role. AWS STS uses a default global endpoint located in the US-East Northern Virginia region at <https://sts.amazonaws.com>; you can also choose to make STS API calls to other AWS regions using a regional endpoint if faster responses are needed. Temporary security credentials are, indeed, temporary, whereas access credentials linked to an IAM user are permanent. Temporary credentials are provided only when access to AWS resources is requested using a role.

The action in the policy can be defined as **AssumeRole**, **AssumeRoleWithSAML**, or **AssumeRoleWithWebIdentity**, as shown in [Figure 3-35](#).

## admin\_access

### Summary

Creation date	ARN
May 08, 2019, 16:17 (UTC-04:00)	<a href="#">arn:aws:iam::313858614000:role/admin_access</a>
Last activity	Maximum session duration
None	1 hour

Permissions    **Trust relationships**    Tags    Access Advisor    Revoke sessions

### Trusted entities

Entities that can assume this role under specified conditions.

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Principal": {  
7         "AWS": "arn:aws:iam::618143137686:root"  
8       },  
9       "Action": "sts:AssumeRole",  
10      "Condition": {}  
11    }  
12  ]  
13 }
```

**Figure 3-35** Trusted Entities in Trust Policy

For either of these actions, STS is called. After verification, STS returns temporary credentials (access key, secret access key, and security token), which are valid for 1 hour by default. You can edit the maximum role session duration to control the exact length of time the assigned security credentials are valid (1 to

36 hours), or a custom length of time can be defined. The advantages of using STS to provide temporary credentials for accessing AWS services are as follows:

- There's no need to rotate security credentials; STS performs credential rotation when temporary credentials are renewed.
- Applications use temporary credentials when they're hosted on EC2 instances with assigned roles, so there is no need for IAM user account credentials and passwords to be embedded in the application.
- STS manages and secures temporary credentials.
- Access to AWS resources can be defined without requiring a full IAM user account.
- Active sessions can be revoked at any time using the IAM dashboard, as shown in [Figure 3-36](#).

**Key  
Topic**

The screenshot shows the AWS IAM 'Revoke sessions' tab selected. Below it, a section titled 'Immediately revoke all active sessions' contains a note about attaching an inline policy named 'AWSRevokeOlderSessions'. A large button labeled 'Revoke active sessions' is present. Below the button, a code snippet shows the JSON structure of the inline policy.

```
1- {
2-     "Version": "2012-10-17",
3-     "Statement": [
4-         {
5-             "Effect": "Deny",
6-             "Action": [
7-                 "*"
8-             ],
9-             "Resource": [
10-                 "*"
11-             ],
12-             "Condition": {
13-                 "DateLessThan": {
14-                     "aws:TokenIssueTime": "[policy creation time]"
15-                 }
16-             }
17-         }
18-     ]
19- }
```

**Figure 3-36** Revoke Active Sessions

## IAM Best Practices



There are several best practices you should consider following when managing user security with IAM:

- **Root account:** Be careful with the AWS root account password. Don't create such a complicated password that you can't remember it and have to write it down. When you need access to the root account, reset the password. In addition, always enable MFA on a root account. Make sure your access keys for the root account have been deleted. You can check whether you have active access keys for your root account by logging on as the root user, opening the IAM console, and making sure the root account access keys have been removed (see [Figure 3-37](#)).

## Your Security Credentials

Use this page to manage the credentials for your AWS account. To manage credentials for AWS Identity and Access Management (IAM) users, use the [IAM Console](#). To learn more about the types of AWS credentials and how they're used, see [AWS Security Credentials](#) in AWS General Reference.

▲ Password

▲ Multi-factor authentication (MFA)

▼ Access keys (access key ID and secret access key)

Use access keys to make programmatic calls to AWS from the AWS CLI, Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of 50 access keys per account. For your protection, you should never share your secret keys with anyone. As a best practice, we recommend frequent key rotation. If you lose or forget your secret key, you cannot retrieve it. Instead, create a new access key and make the old key inactive. [Learn more](#)

Created	Access Key ID	Last Used	Last Used Region
<a href="#">Create New Access Key</a>			

Root user access keys provide unrestricted access to your entire AWS account. If you need long-term access keys, we recommend creating a new IAM user instead. [Learn more](#)

**Figure 3-37** Properly Set Up Root Account

- **Individual IAM users and groups for administration:** Even when creating single IAM users, consider placing them in an IAM group. At some point, each single user's duties may need to be assumed by someone else due to holidays or illness. It's much easier to add a new IAM user to an existing IAM group than to manage separate individual IAM users.
- **Permissions:** Grant least privileges when assigning IAM permissions. Take the time to get proficient at deploying IAM management policies. If necessary, create custom IAM policies for specific administrative access. Remember that most IAM accounts are administrator accounts. The goal should be to use IAM roles wherever possible because roles use controlled access with temporary credentials that are assigned and completely managed by STS.
- **Groups:** If possible, don't manage by individual IAM users; instead, manage by delegating access using IAM groups.
- **Conditions:** Consider restricting access with additional policy conditions. Consider adding a mandatory IP address range for administrators who need to perform administrative tasks and force authentication and access from a specific range of IP addresses.
- **CloudTrail logs:** Create a custom CloudTrail trail that saves all API calls and authentications from all AWS regions to a defined S3 bucket forever.

- **Passwords:** Make sure to create a strong password policy that matches corporate requirements.
- **Security credential rotation:** Consider rotating the security credentials on a timeline that matches the password change for each account. Even better, redesign your IAM security to use IAM roles for administrative tasks to ensure temporary credentials are used and managed by STS/AWS.
- **MFA:** Enable MFA for IAM users, including the root user of each AWS account. At the very least, use a software-based security code generator such as Google Authenticator or Authy.
- **Use IAM roles for application servers:** Use IAM roles to share temporary access to AWS resources for applications hosted on EC2 instances. Let AWS and STS manage application credentials.



## IAM Security Tools

Various security utilities and tools are available to make your job of managing IAM security easier. The following are tools to know for the SAA-C03 exam:

- **Credential Report:** From the IAM dashboard or using the AWS CLI, request and download a comma-separated values (CSV) report that lists the current status of IAM users in an AWS account (see [Figure 3-38](#)). Details include the status of the access keys (for example, usage, last used service, key rotation, passwords enabled/disabled, last time used, last changed, and MFA status). The information provided by the report is within the most recent 4-hour time status).

**Credential Report**

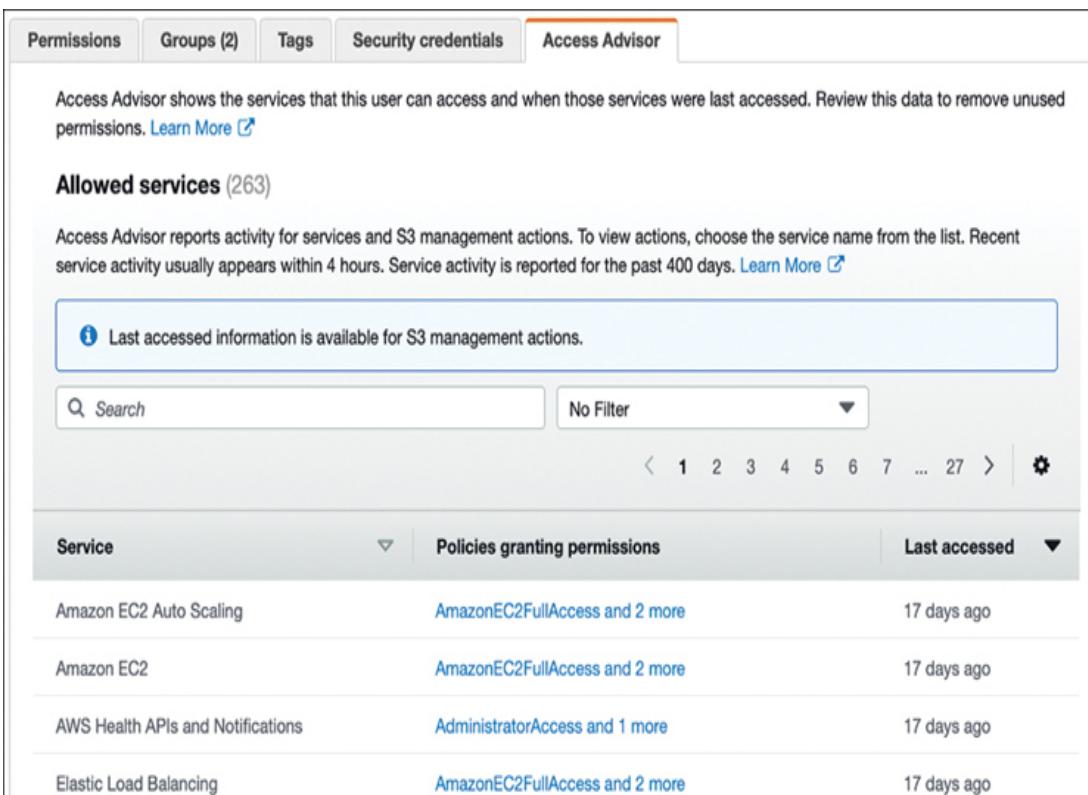
Click the button to download a report that lists all your account's users and the status of their various credentials. to four hours. For more information see the [documentation](#).

**Download Report**

	A	B	C	D	E	F	G	H	I	J	K
1	user	arn	user_creation_date	password_expiration_date	password_last_changed	password_never_expired	mfa_active	access_key_1_status	access_key_1_last_used_date	access_key_2_status	access_key_2_last_used_date
2	<root_account>	arn:aws:iam	2011-05-23T	not_supported	2021-01-01T	not_supported	not_supported	TRUE	FALSE	N/A	N/A
3	bart	arn:aws:iam	2021-01-01T	TRUE	no_informat	2021-01-01T	2021-02-15T	FALSE	TRUE	2021-01-01T	N/A
4	biff	arn:aws:iam	2018-04-16T	TRUE	no_informat	2018-04-16T	2018-05-31T	FALSE	TRUE	2018-04-16T	N/A
5	bobby	arn:aws:iam	2018-04-12T	TRUE	no_informat	2018-04-12T	2018-05-27T	FALSE	TRUE	2018-04-12T	N/A
6	brenda	arn:aws:iam	2018-03-27T	TRUE	no_informat	2018-03-27T	2018-05-11T	FALSE	TRUE	2018-03-27T	N/A
7	brock	arn:aws:iam	2020-09-08T	TRUE	no_informat	2020-09-08T	2020-10-23T	TRUE	TRUE	2020-09-08T	N/A
8	cathy	arn:aws:iam	2020-03-02T	TRUE	no_informat	2020-03-02T	2020-04-16T	FALSE	TRUE	2020-03-02T	N/A
9	clint	arn:aws:iam	2018-10-12T	TRUE	no_informat	2018-10-12T	2018-11-26T	FALSE	TRUE	2018-10-12T	N/A

**Figure 3-38** Credential Report

- **Access Advisor:** Reports can be generated to display the last time an IAM user or role accessed an AWS service. View reports for each IAM entity by first selecting the IAM user, group, or role, selecting the Access Advisor tab, and then viewing the contents of the Access Advisor tab, as shown in [Figure 3-39](#).



The screenshot shows the 'Access Advisor' tab selected in the top navigation bar. A message states: 'Access Advisor shows the services that this user can access and when those services were last accessed. Review this data to remove unused permissions.' Below this, a section titled 'Allowed services (263)' provides information about service activity and includes a note: 'Last accessed information is available for S3 management actions.' A search bar and filter dropdown are present. The main table lists services, policies granting permissions, and the last access time.

Service	Policies granting permissions	Last accessed
Amazon EC2 Auto Scaling	AmazonEC2FullAccess and 2 more	17 days ago
Amazon EC2	AmazonEC2FullAccess and 2 more	17 days ago
AWS Health APIs and Notifications	AdministratorAccess and 1 more	17 days ago
Elastic Load Balancing	AmazonEC2FullAccess and 2 more	17 days ago

**Figure 3-39** Access Advisor Details

- **Policy Simulator:** After you've created your first policy, you might get lucky and have it work right away. If you are using the pre-created managed policies provided by AWS, they will

usually work. However, a custom policy might not work as expected. Fortunately, Amazon has a simulator called the IAM Policy Simulator that you can use to test your policies (see [Figure 3-40](#)). The simulator evaluates your policy using the same policy evaluation engine that would be used if real IAM policy requests were being carried out.

The screenshot shows the IAM Policy Simulator interface. On the left, there's a sidebar titled 'Policies' with a 'Selected user: cynthia' dropdown, a 'Create New Policy' button, and a 'IAM Policies' section containing three checked checkboxes: 'IAMUserChangePassword', 'AdministratorAccess', and 'AmazonEC2FullAccess'. A 'Filter' input field is also present. To the right, the main area is titled 'Policy Simulator' with tabs for 'AWS Certificate Manager' and 'Action Settings and Results'. Under 'Action Settings and Results', it shows '1 Action(s) sele...' and '1 actions selected. 0 actions not simulated. 1 actions allowed. 0 actions denied.' Below this is a table with columns: Service, Action, Resource Type, Simulation Resource, and Permission. One row is visible: 'AWS Certificate Manager GetCertificate certificate \* allowed'.

**Figure 3-40** IAM Policy Simulator

With the Policy Simulator, you can test IAM policies that are attached to IAM users, groups, or roles within an AWS account. You can select one or all security policies that are attached, and you can test all actions for what is being allowed or denied by the selected IAM policies. You can even include conditions such as the IP address that the request must come from. Both identity-based and resource-based policies can be tested with the Policy Simulator.

**Key  
Topic**

## IAM Cheat Sheet

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of IAM:

- New IAM users are created with no access to any AWS services.
- IAM users can be assigned access keys, passwords, and multi-factor authentication.
- By using identity federation in conjunction with IAM roles, you can enable secure access to resources without needing to create an IAM user account.
- IAM is a global service that is not restricted to a single AWS region.
- IAM roles use security credentials provided by AWS STS that provide temporary access to AWS services and resources.
- Temporary security credentials include an AWS access key, a secret access key, and a security token.
- Each AWS account root account has full administrative permissions that cannot be restricted.

- IAM roles define a set of permissions for allowing or denying actions to AWS services.
- IAM roles are assumed by trusted identities.
- IAM roles allow you to delegate access permissions to AWS resources without requiring permanent credentials.
- There are no credentials assigned to an IAM role.
- IAM roles have two policies:
  - The permissions policy defines the permissions required for the role.
  - The trust policy defines the trusted accounts that are allowed to assume the role.



## AWS Identity Center

AWS Identity Center, the successor to AWS Single Sign-On, is a cloud-based SSO service that manages access and permissions to third-party cloud applications and applications that support SAML 2.0 for AWS accounts contained in AWS Organizations. AWS Identity Center integrates with AWS Organizations and enumerated AWS accounts supporting the following features:

- Provides SSO access to cloud applications for AWS accounts

- Provides SSO access to AWS applications such as SageMaker
- Provides SSO access to EC2 Windows desktops
- Provides SSO access to IAM users and groups, AWS-Managed Microsoft AD directory users, and external identity providers
- Provides SSO access to many popular cloud-hosted applications (Salesforce, Box, Office 365)

To get started with AWS Identity Center, complete the following steps:

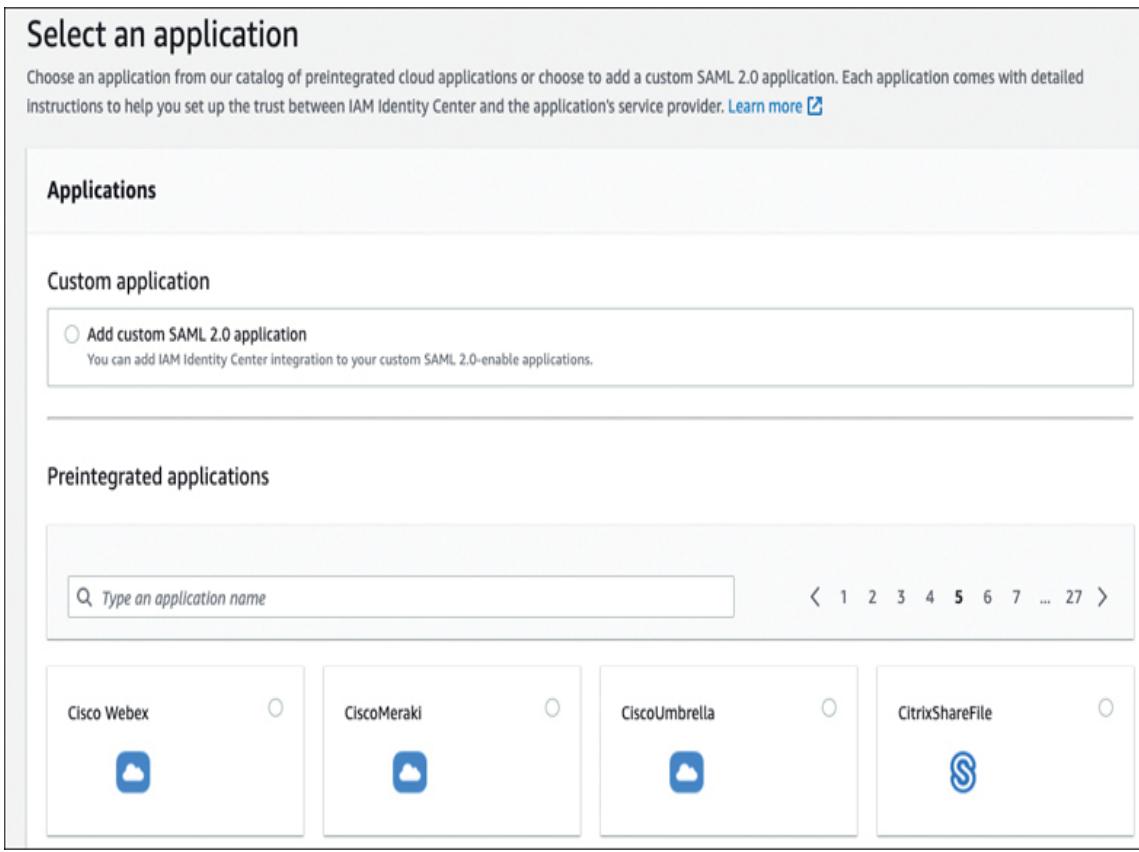
**Step 1.** AWS Organizations must first be deployed.

**Step 2.** Sign in using the AWS Organizations Management account credentials, which are required to enable AWS Identity Center.

**Step 3.** Choose the identity store that will have access to the AWS Identity Center user portal.

**Step 4.** After opening the AWS Identity Center console for the first time, enable the AWS Identity Center service.

**Step 5.** Add and configure applications that are to be integrated with AWS Identity Center, as shown in [Figure 3-41](#).



**Figure 3-41** AWS Identity Center Cloud Applications

## AWS Organizations

AWS Organizations enables centralized policy-based management for multiple AWS accounts that are grouped together in a tree structure. If you're lucky enough to not yet have multiple AWS accounts, you can look at AWS Organizations as a great starting point, especially if you know you're eventually going to have multiple AWS accounts to manage.

The first step to carry out with AWS Organizations is to create your initial organization with a specific AWS account; this account will henceforth be known as the *management account* (see [Figure 3-42](#)). The management account sits at the root of your AWS organization's tree.

AWS Organizations > AWS accounts > Root

## Root

Root is the parent organizational unit (OU) for all accounts and other OUs in your organization. When you apply a policy to the root, it applies to every OU and account in the organization. [Learn more](#)

Root details		
ID	r-a2b6	
ARN	arn:aws:organizations::313858614000:root/o-bq5yhpe6ls/r-a2b6	
Enabled policy types ( <a href="#">manage policy types</a> )	Service control policies	
<a href="#">Children</a> <a href="#">Tags</a> <a href="#">Policies</a>		

Children		Actions ▾
These are organizational units and AWS accounts attached directly to Root.		
Organizational structure	Account created/Joined date	
▶ <input type="checkbox"/> <input type="checkbox"/> Canada	ou-a2b6-le3rmvek	
▶ <input type="checkbox"/> <input type="checkbox"/> Sales	ou-a2b6-0vr4s2ut	
▶ <input type="checkbox"/> <input type="checkbox"/> Sandbox	ou-a2b6-ewmhf67l	

**Figure 3-42** AWS Organizations

---

## Note

The management account is also called the *payer account* because it is responsible for all the charges carried out by all the AWS accounts that are nested within AWS Organizations. AWS Organizations includes, by default, consolidated billing.

---

Using AWS Organizations, at the root, you can create new AWS accounts or add existing AWS accounts. All additional AWS accounts added to AWS Organizations are defined as member accounts. After grouping your AWS accounts, you can then apply security control policies to them. As introduced earlier in this chapter, the policies that can be applied to AWS Organizations are called service control policies (SCPs); these are permission boundaries that help define the effective permissions of applied IAM policies. If an SCP and an IAM policy assigned to a specific AWS account IAM user allow the same AWS service actions in both policy documents—that is, if the settings match—then the actions are allowed.

Within AWS Organizations, the AWS accounts can be organized into groupings called *organizational units (OUs)*, as shown in [Figure 3-43](#). OUs can be nested to create a tree-like hierarchy that meets your organization's needs and requirements. Nested OUs inherit SCPs from the parent OU and specific policy

controls that can be applied directly to any OU. SCPs can be defined for an entire AWS organization, for specific OUs, or for specific AWS accounts located within an OU.

## AWS accounts

Add an AWS account

The accounts listed below are members of your organization. The organization's management account is responsible for paying the bills for all accounts in the organization. You can use the tools provided by AWS Organizations to centrally manage these accounts. [Learn more](#)

**Organization** Actions ▾

Organizational units (OUs) enable you to group several accounts together and administer them as a single unit instead of one at a time.

Find AWS accounts by name, email, or account ID. Find an OU by the exact OU ID. Hierarchy List

Organizational structure	Account created/joined date
▼ <input type="checkbox"/> <input type="checkbox"/> Root r-a2b6	
▶ <input type="checkbox"/> <input type="checkbox"/> Canada ou-a2b6-le3rmvek	
▶ <input type="checkbox"/> <input type="checkbox"/> Sales ou-a2b6-0vr4s2ut	
▶ <input type="checkbox"/> <input type="checkbox"/> Sandbox ou-a2b6-ewmhf67l	
▶ <input type="checkbox"/> <input type="checkbox"/> Security ou-a2b6-ecfl1bo	
▶ <input type="checkbox"/> <input type="checkbox"/> USA ou-a2b6-1mmdwsj2	

**Figure 3-43** OUs in AWS Organizations

## AWS Organizations Cheat Sheet

**Key  
Topic**

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of AWS Organizations:

- An AWS organization is a collection of AWS accounts organized into a hierarchy that can be managed centrally.
- Each AWS account in an organization is designated as a member account located in a container. There is no technical difference between the master account and a member account other than its location.
- AWS Organizations supports consolidated billing.
- AWS Resource Access Manager can be used to share resources within the organization tree.
- Service control policies (SCPs) can be applied to AWS accounts or OUs contained within the AWS organization controlling access to AWS resources and services.
- AWS CloudTrail can be activated across all AWS accounts in the organization and cannot be turned off by member accounts.
- An organizational unit contains one or more AWS accounts within the AWS organizational tree.

- Security tools (AWS IAM, AWS Config, AWS Control Tower) can manage the needs and requirements of the AWS accounts that are members of the same AWS organization.
- AWS Cost Explorer can be used to track costs across accounts.

## AWS Resource Access Manager



AWS Resource Access Manager (RAM) allows you to centrally manage resources across AWS accounts and AWS Organizations.

AWS RAM allows you to share selected AWS resources hosted within a single AWS account with other AWS accounts. If you are using AWS Organizations, AWS RAM can also be used to share AWS resources between AWS accounts that are members of the same AWS Organization.

AWS RAM can share application or database servers between different AWS accounts instead of having to create duplicate resources.

To share resources using AWS RAM, first create a resource share (see [Figure 3-44](#)), configure the permissions to use for the

resource, and select the principals that will have access to the resource.

With AWS RAM, the first task is to decide which resources that you own that you want to share. Next, you need to decide which principals to share the resource with; resource principals can be AWS accounts, OUs, IAM users, or the entire AWS organization.

Resources - optional					
Choose the resources to add to the resource share					
Select resource type					
Subnets					
<input type="text"/> Filter by attributes or search by keyword					
ID	Name	VPC ID	Availability zone	Availability zone ID	
<input checked="" type="checkbox"/>	subnet-35441b18	Private Subnet	vpc-c753f9a2	us-east-1d	use1-az2
<input type="checkbox"/>	subnet-265f5f7c	Private Subnet 2	vpc-6d30d915	us-east-1b	use1-az6
<input type="checkbox"/>	subnet-b03ff9fb	Private Subnet 1	vpc-6d30d915	us-east-1a	use1-az4

**Figure 3-44** Sharing Subnets with AWS RAM

If your AWS account is a member of an AWS organization, once sharing is enabled, any selected resource principal will be granted access to any resources shared by a resource share, as shown in [Figure 3-45](#). If AWS Organizations is not deployed, the separate AWS account will receive an invitation from the AWS

owner account that has created the resource share to join the resource share—after accepting the invitation, the AWS account will have access to the shared resource.

**Principals - optional**

**Allow sharing with anyone**  
You can share resources with any AWS accounts, roles, and users. If you are in an organization, you can also share with the entire organization or organizational units in that organization.

**Allow sharing**  
You can share re accounts, roles, a

**Principals**  
You can add multiple principals of different types. To display and select principals from a hierarchical view of your organiz

**Display organizational structure**

<input type="checkbox"/>	Name	ID
<input type="checkbox"/>	<input type="checkbox"/> [ ]	o-bq5yhpe6ls
<input type="checkbox"/>	<input type="checkbox"/> Canada	ou-a2b6-le3rmvek
<input type="checkbox"/>	<input type="checkbox"/> Sales	ou-a2b6-0vr4s2ut
<input type="checkbox"/>	No organizational units or accounts exist	
<input type="checkbox"/>	<input type="checkbox"/> Sandbox	ou-a2b6-ewmhf67l
<input type="checkbox"/>	No organizational units or accounts exist	
<input type="checkbox"/>	<input type="checkbox"/> Security	ou-a2b6-ecfllbo
<input type="checkbox"/>	Audit	152568481382
<input type="checkbox"/>	Log Archive	444784811587
<input type="checkbox"/>	<input type="checkbox"/> USA	ou-a2b6-1mmdwsj2

**Figure 3-45** Selecting Principals to Grant Access

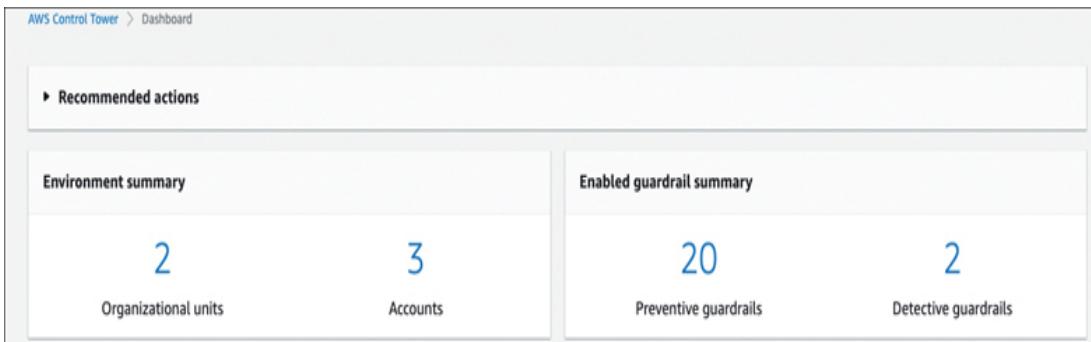
The tasks that can be performed with the shared resource depends on the type of resource shared and the IAM security policies and service control policies that may have been applied. AWS resources that can be shared with the AWS Resource Access Manager include Aurora DB clusters, Capacity reservations, Dedicated hosts, Glue catalogs, Image Builder images, AWS Outposts, and Transit Gateways.



## AWS Control Tower

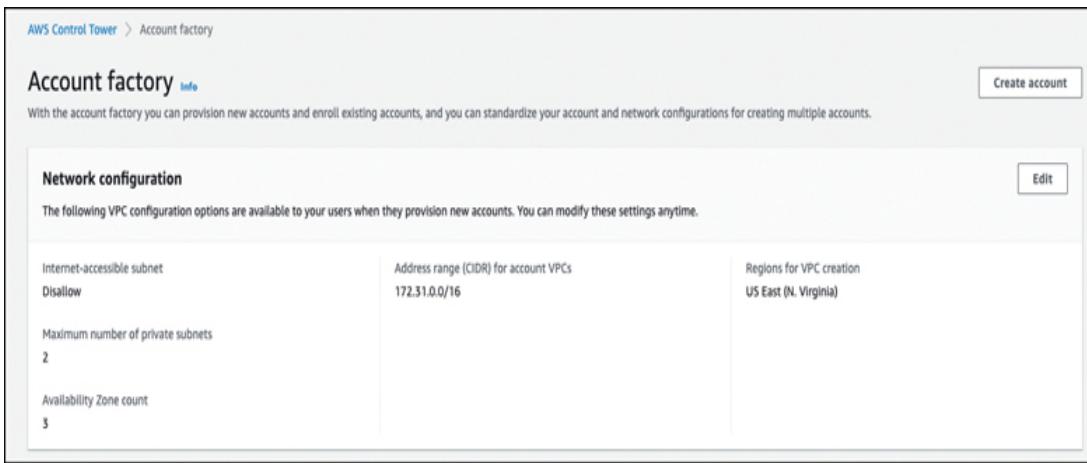
AWS Control Tower automates the creation and governance of a secure, multi-account AWS environment using AWS Organizations, as shown in [Figure 3-46](#). AWS Control Tower also automates the creation of a landing zone for onboarding AWS accounts using prebuilt blueprints that follow suggested best practices for configuring a default identity, federated access, and account structure. Deploying AWS Control Tower deploys and configures and integrates the following AWS services:

- AWS Organizations is deployed creating a multi-AWS account structure.



**Figure 3-46** AWS Control Tower Landing Zone

- AWS Organization's SCPs are deployed to prevent unwanted configuration changes.
- Federated access is enabled using AWS Identity Center.
- Central log archiving using AWS CloudTrail and AWS Config is stored in Amazon S3.
- Security audits are enabled across all AWS accounts using AWS Identity Center and Identity and Access Management.
- **Account Factory:** The account factory automates the provisioning of new AWS accounts in the deployed AWS Organization tree. Preapproved network configurations and AWS region selections can also be defined as the mandatory network baseline for all new AWS accounts, as shown in [Figure 3-47](#).



**Figure 3-47** Account Factory Preapproved Network Configurations

- **Guardrails:** Guardrails are used to provide ongoing governance of the AWS environment by preventing deployment of resources that don't follow prescribed policies. Guardrails prevent deployment of resources that don't match your rules. Guardrails can also be detective in nature by continually monitoring the resources that are deployed for nonconformance. Guardrails are deployed using an AWS CloudFormation script that establishes the configuration baseline. SCPs create preventive guardrails that prevent unwanted infrastructure changes. Detective guardrails are created and enforced using AWS Config rules. There are also mandatory and optional guardrails that can be leveraged, as shown in [Figure 3-48](#). For example, organizations can mandate that any changes to logging

configuration for Amazon S3 bucket policies can be set to disallowed. An example of an optional guardrail that can be enabled is detecting whether MFA is enabled for the root user.

### Key Topic

Guardrails <small>Info</small>			
Name	Guidance	Category	Behavior
Disallow deletion of log archive	Mandatory	Audit logs	Prevention
Disallow Changes to Encryption Configuration for Amazon S3 Buckets	Elective	Audit logs	Prevention
Disallow Changes to Logging Configuration for Amazon S3 Buckets	Elective	Audit logs	Prevention
Detect public read access setting for log archive	Mandatory	Audit logs	Detection
Detect public write access setting for log archive	Mandatory	Audit logs	Detection

**Figure 3-48** AWS Control Tower Guardrails

## Exam Preparation Tasks

As mentioned in the section “[How to Use This Book](#)” in the Introduction, you have a couple of choices for exam preparation: the exercises here, [Chapter 16](#), “[Final](#)

Preparation,” and the exam simulation questions in the Pearson Test Prep Software Online.

## Review All Key Topics

Review the most important topics in the chapter, noted with the Key Topic icon in the margin of the page. [Table 3-3](#) lists these key topics and the page number on which each is found.



**Table 3-3** [Chapter 3](#) Key Topics

Key Topic Element	Description	Page Number
<a href="#">Figure 3-2</a>	AWS Config Service-Linked Roles Defined by IAM	81
Section	IAM Policy Definitions	81
Paragraph	IAM authentication	82

<b>Key Topic Element</b>	<b>Description</b>	<b>Page Number</b>
<a href="#"><u>Figure 3-3</u></a>	IAM User Account Access Keys	83
<a href="#"><u>Figure 3-4</u></a>	An Amazon AWS Resource Name (ARN)	85
<a href="#"><u>Figure 3-6</u></a>	Policy Evaluation Logic	87
<a href="#"><u>Figure 3-8</u></a>	Root User Logon	89
<a href="#"><u>Figure 3-10</u></a>	Access Keys Required for CLI Operation	92
<a href="#"><u>Figure 3-12</u></a>	Using Custom URL for IAM Users	95
<a href="#"><u>Figure 3-15</u></a>	Creating an Additional Access Key Manually	98

<b>Key Topic Element</b>	<b>Description</b>	<b>Page Number</b>
Section	Using Multi-Factor Authentication	99
<u>Figure 3-18</u>	Managed Policies	101
Section	Resource-Based Policies	102
<u>Figure 3-20</u>	Identity and Resource Policies Working Together	103
Section	Policy Elements	106
<u>Example 3-2</u>	IAM Policy for Performing Administrative Tasks	110
Section	Additional Policy Control Options	110

<b>Key Topic Element</b>	<b>Description</b>	<b>Page Number</b>
Section	AWS Organizations Service Control Policies	112
Section	IAM Policy Versions	115
<u>Table 3-2</u>	Conditional Elements	116
Section	IAM Roles	118
Section	When to Use IAM Roles	119
<u>Figure 3-29</u>	Attaching an IAM Role to an EC2 Instance	120
Section	Web Identity Federation	121
<u>Figure 3-31</u>	Using Cognito for Mobile User Authentication	122

<b>Key Topic Element</b>	<b>Description</b>	<b>Page Number</b>
<u><a href="#">Figure 3-34</a></u>	Using the Switch Role Option for Cross-Account Access	126
Section	AWS Security Token Service	126
<u><a href="#">Figure 3-36</a></u>	Revoke Active Sessions	128
Section	IAM Best Practices	128
Section	IAM Security Tools	130
Section	IAM Cheat Sheet	132
Section	AWS Identity Center	132
Section	AWS Organizations Cheat Sheet	136

<b>Key Topic Element</b>	<b>Description</b>	<b>Page Number</b>
Section	AWS Resource Access Manager	136
Section	AWS Control Tower	138
<u>Figure 3-48</u>	AWS Control Tower Guardrails	139

## Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

Identity and Access Management (IAM)

multi-factor authentication (MFA)

externally authenticated user

condition

[access key](#)

[IAM group](#)

[password policies](#)

[IAM role](#)

## Q&A

The answers to these questions appear in [Appendix A](#). For more practice with exam format questions, use the Pearson Test Prep Software Online.

- 1.** How can you tell when you're using the root account?
- 2.** What is the best way to give an application secure access to AWS services?
- 3.** What is the advantage of using a resource-based policy instead of an identity-based policy to protect an S3 bucket?
- 4.** What is the best method for controlling access to AWS resources?
- 5.** Why should inline policies be discouraged in most cases?

- 6.** Which tool can you use to check your policies for proper functionality?
- 7.** How can AWS Organizations help you manage multiple AWS accounts?
- 8.** What security components are required to run a script from the command-line interface?

# Chapter 4

## Designing Secure Workloads and Applications

This chapter covers the following topics:

- [Securing Network Infrastructure](#)
- [Amazon Cognito](#)
- [External Connections](#)
- [Amazon GuardDuty](#)
- [Amazon Macie](#)
- [Security Services for Securing Workloads](#)

This chapter covers content that's important to the following exam domain and task statement:

### **Domain 1: Design Secure Architectures**

Task Statement 2: Design secure workloads and applications

Workload and application security at AWS refers to the measures and controls that are implemented to protect the data and associated cloud services used to process, store, and transmit data. This includes implementing security controls and

practices to protect against potential threats and vulnerabilities that could compromise the security of workloads.

To properly secure workload network infrastructure in the cloud, organizations must design and deploy subnets and route tables, security groups (SG), and network access control lists (ACLs) to protect workload infrastructure hosted on subnets in each virtual private cloud (VPC). There are also security services to consider deploying at each edge location, including the AWS Web Application Firewall (WAF), AWS Shield Standard, and AWS Shield Advanced, to help protect workloads that are exposed to the Internet.

Workload security can also utilize a combination of security controls provided by the AWS, such as utilizing Amazon Cognito, which provides authentication, authorization, and user management for your web and mobile applications. There are also several AWS security services that can assist in securing the associated workload, including Amazon Macie, Amazon GuardDuty, AWS CloudTrail, AWS Secrets Manager, Amazon Inspector, and AWS Trusted Advisor.

### **“Do I Know This Already?”**

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or

jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your assessment of your knowledge of the topics, read the entire chapter. [Table 4-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A](#), “[Answers to the ‘Do I Know This Already?’ Quizzes and Q&A Sections.](#)”

**Table 4-1** “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Securing Network Infrastructure	1, 2
Amazon Cognito	3, 4
External Connections	5, 6
Amazon GuardDuty	7, 8
Amazon Macie	9, 10
Security Services for Securing Workloads	11, 12

---

## Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment.

Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

---

**1.** Which route table is associated with each new subnet at creation?

1. Custom route table
2. None, because route tables are not automatically associated
3. The main route table
4. The network access control list

**2.** What is a security group's job?

1. To deny incoming and outgoing traffic
2. To control incoming and outgoing traffic
3. To block incoming and outgoing traffic
4. To explicitly deny incoming and outgoing traffic

**3.** What services are analyzed by Amazon GuardDuty?

1. Amazon EFS and FSx for Windows File Server logs
2. AWS Route 53 logs and Amazon VPC flow logs
3. Amazon Inspector logs and AWS Config rules
4. Amazon RDS logs and Amazon EC2 system logs

**4.** What AWS service is used to automate remediation of Amazon GuardDuty issues?

1. Amazon CloudTrail
2. Amazon CloudWatch
3. AWS Lambda
4. Amazon Route 53

**5.** What VPN service needs to be installed before connecting to a VPC?

1. AWS Direct Connect
2. AWS Customer Gateway
3. Virtual Private Gateway
4. AWS VPN Cloud Hub

**6.** What type of network connection is an AWS Direct Connect connection?

1. Public
2. Single-mode fiber
3. VPN
4. IPsec

**7.** What type of data records are analyzed by Amazon Macie?

1. Amazon S3 buckets
2. Amazon EFS
3. Amazon S3 Glacier
4. Amazon FSx for Windows File Server

**8.** What process is used by Amazon Macie to begin a data analysis?

1. Administrator
2. Schedule
3. Job
4. Task

**9.** What does AWS Cognito use to authenticate end users to a user pool?

1. Username/password
2. Username/SNS
3. Username/email/phone number

#### 4. MFA and password

**10.** What does Amazon Cognito require to authenticate mobile application users?

1. User pool
2. Identity pool
3. Certificates
4. MFA

**11.** Which of the following can you use to retain AWS CloudTrail events permanently?

1. AWS Lambda function
2. A custom trail
3. AWS Step Function
4. None of these; events can be retained for only 90 days

**12.** Which of the following does Amazon Inspector evaluate?

1. Amazon S3 buckets
2. Amazon Elastic Container Service
3. Amazon EC2 instances
4. Amazon Relational Database Service

## Foundation Topics

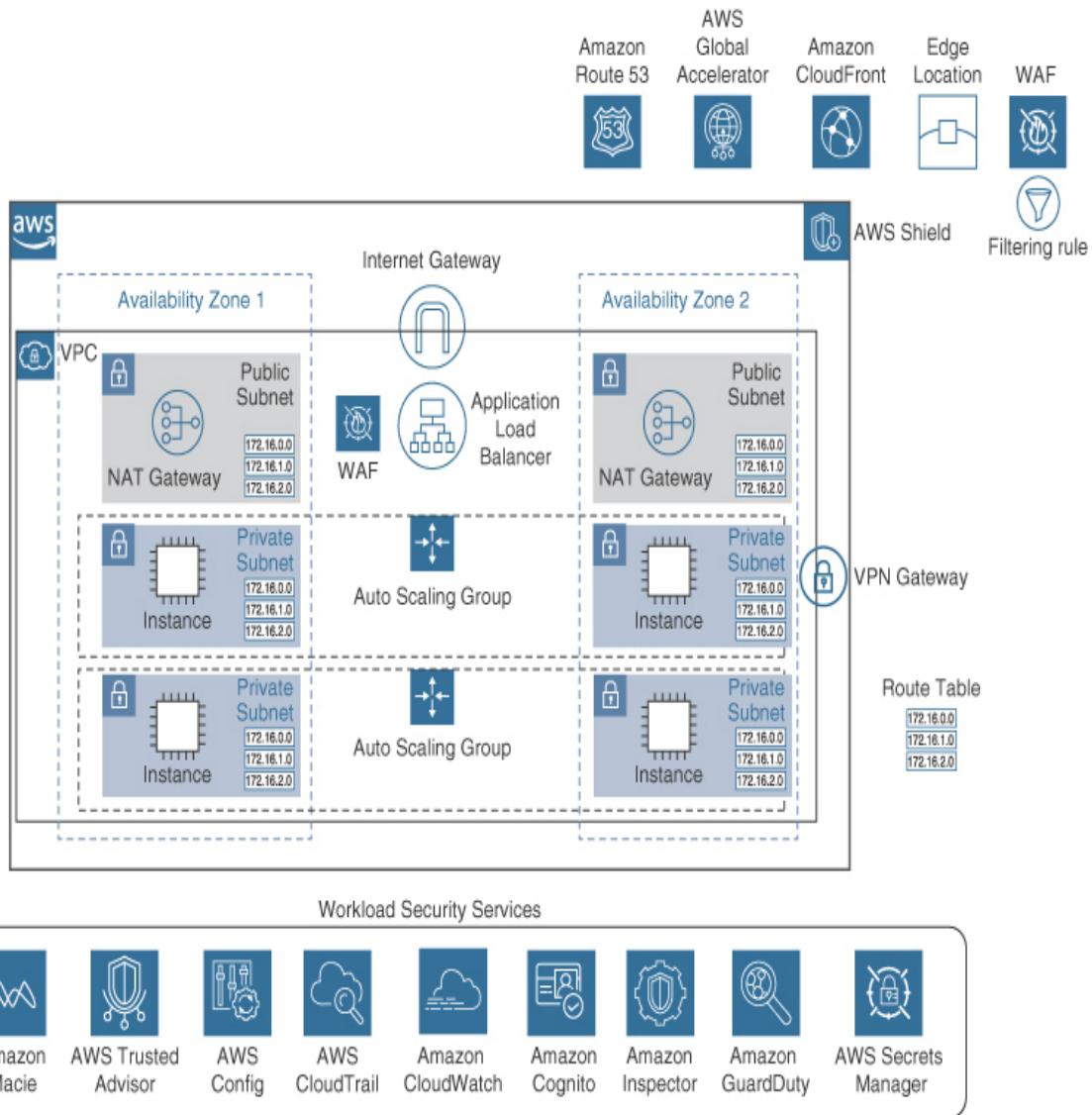
## Securing Network Infrastructure

Workloads can be protected with a variety of AWS security services. Connections to workloads running at AWS can utilize both public and private connections using one or more of the following AWS networking services (see [Figure 4-1](#)):

- **Internet connections (HTTPS/HTTP) to public-facing workloads:** An Internet gateway must be attached to the VPC hosting the workload, and the VPC's security groups and network ACLs must be configured to allow the appropriate traffic to flow through the subnets, load balancer, and web servers.
- **AWS Direct Connect:** Establish a dedicated network connection from your on-premises data center to your VPC (Virtual Private Cloud) at AWS.
- **An AWS VPN (Virtual Private Network) connection:** Provide a secure encrypted connection between an on-premises network and VPC at AWS. VPN connections allow access to your AWS resources and workloads as if they were on your on-premises network.
- **Edge locations for accessing cached data records using Amazon CloudFront, Amazon's content delivery network (CDN):** Edge locations are located at the edge of the AWS cloud and serve content to users more quickly and efficiently.

- **AWS Global Accelerator network:** Use Amazon's global network of edge locations to improve the performance of Internet applications routing traffic from users to the optimal AWS endpoint.

**Key  
Topic**



**Figure 4-1** Connections and Security Services

## Networking Services Located at Edge Locations

Edge locations are located at the edge of the AWS regions and are used to serve content to users more quickly and efficiently. These essential AWS services are located at each edge location:

- **Amazon Route 53:** Amazon-provided DNS services for resolving queries to Amazon CloudFront and AWS Global Accelerator deployments. Additional details on Route 53 operation can be found in [Chapter 7, “Designing Highly Available and Fault-Tolerant Architecture.”](#)
- **AWS Shield:** Provides protection against distributed denial of service attacks (DDoS).
- **AWS Web Application Firewall (WAF):** Protect web applications from common web exploits that could affect application availability, compromise security, or consume excessive resources. AWS WAF enables you to create rules that block, allow, or monitor web requests based on conditions.
- **Amazon CloudFront:** CloudFront serves cached static and dynamic content from edge locations rather than from the origin data location (S3 bucket or application server), which can reduce the amount of time it takes to access static, dynamic, or streaming content. Additional details on CloudFront operation can be found in [Chapter 11, “High-Performing and Scalable Networking Architecture.”](#)



## AWS Shield (Standard and Advanced)

What if a malicious request, DDoS attack, or a malicious bot attempts to enter an AWS edge location and attack a public-facing application? AWS Shield Standard protection protects AWS infrastructure and ingress paths at each edge location for all AWS customers. AWS Shield runs at each edge location, providing basic DDoS protection for known Layer 3 and Layer 4 attacks using AWS Web Application Firewall rules deployed and managed by AWS.

If organizations don't have the required expertise needed to solve ongoing security exploits that are attacking workloads hosted at AWS, they can contract with AWS experts to assist with real-time custom protection, known as AWS Shield Advanced, a paid version of AWS Shield that utilizes an expert AWS DDoS response team with a 15-minute SLA response protecting your workload components (Amazon EC2 instances, AWS Elastic Load Balancers, Amazon CloudFront distributions, AWS Global Accelerator deployments, and Amazon Route 53 resources). After analyzing the situation, the response team creates and applies custom WAF filters to mitigate DDoS attacks and other security issues. All AWS Shield Advanced customers get access to a global threat dashboard (see [Figure 4-2](#)) that displays a sampling of current attacks.

## Global threat dashboard across all AWS customers

The following is a sampling of the most significant attacks that AWS is monitoring and mitigating for customers on Amazon EC2, Amazon CloudFront, Elastic Load Balancing, and Amazon Route 53.

Attack frequency map



**Figure 4-2** Global Threat Dashboard

AWS Shield Advanced also provides *cost protection*, which saves customers money when workload compute resources are required to scale due to illegitimate demand placed on the workload cloud services by a DDoS attack. AWS refunds the additional load balancer, compute, data transfer, and Route 53 query costs that accumulate during the DDoS attack for AWS Shield Advanced customers.

AWS Shield Advanced costs \$3,000 a month with a one-year commitment. AWS WAF and AWS Firewall Manager are included with AWS Shield Advanced at no additional charge.

---

#### Note

Multiple WAF rules can be managed across multiple AWS accounts and workloads using AWS Firewall Manager.

---



## AWS Web Application Firewall (WAF)

For custom workload protection at each edge location, the AWS Web Application Firewall (WAF) provides custom filtering of incoming (ingress) public traffic requests for IPv4 and IPv6 HTTP and HTTPS requests at each edge location, limiting any malicious request from gaining access to AWS cloud infrastructure. AWS WAF rules are created using the AWS Management Console or the AWS CLI. WAF rules are created using conditions combined into a web ACL. WAF rules allow or block, depending on the conditions, as shown in [Figure 4-3](#). WAF rules can be applied to public-facing application load

balancers, Amazon CloudFront distributions, Amazon API gateway hosted APIs, and AWS AppSync. To create a WAF rule, specify the conditions that will trigger the rule, such as the source IP address or the content of a web request. Next, specify the action that the rule should take when the conditions are met, such as blocking or allowing the request. Behaviors and conditions can be used to create custom rules that meet the specific security requirements for your web applications. WAF supports the following behaviors:

- **IP addresses:** Create rules that allow or block requests based on the source or destination IP address.
- **HTTP methods:** Create rules that allow or block requests based on the HTTP method used in the request, such as blocking all PUT requests.
- **Cookies:** Create rules that allow or block requests based on the presence or absence of cookies in the request, such as a specific cookie that is required for authentication.
- **Headers:** Create rules that allow or block requests based on the contents of the request header.
- **Query strings:** Create rules that allow or block requests based on the contents of the query string.

The screenshot shows the 'Rules' section of the AWS WAF configuration interface. At the top, there are three buttons: 'Edit', 'Delete', and 'Add rules ▾'. Below these, a table lists a single rule:

Name	Capacity	Action
AWS-AWSManagedRulesAnonymousIpList	50	Use rule actions

Below the table, a section titled 'Web ACL rule capacity units used' indicates that 50 out of 1500 WCU units are used. A button labeled '50/1500 WCUs' is present. At the bottom, a section titled 'Default web ACL action for requests that don't match any rules' shows that the 'Allow' option is selected.

**Figure 4-3** Web Application Firewall Rules

AWS WAF provides three types of rules:

- **Regular rules:** Used to specify conditions that must be met in order for the rule to be triggered. For example, you might create a regular rule that blocks requests from a specific IP address or that allows requests that contain a specific string in the query string.
- **Rate-based rules:** Used to limit the rate at which requests are allowed to be made to a web application. For example,

you might create a rate-based rule that allows no more than 600 requests per second from a single IP address.

- **Group rules:** Used to group together multiple regular and rate-based rules, applying them as a single entity. For example, you might create a group rule that combines a regular rule that blocks requests from a specific IP address with a rate-based rule that limits the rate at which requests are allowed. This enables you to apply multiple rules to a web application with a single group rule.

## VPC Networking Services for Securing Workloads

Each VPC provides a number of security features designed to protect workloads running in the AWS cloud. Features covered in this section include route tables, security groups, and network ACLs. It's important to realize that these security features are assigned to a specific VPC when they are created.

### Route Tables

Each route table is used to control subnet traffic using a set of rules, called routes, that determine where network traffic is directed within each VPC. Workloads running on virtual servers or containers are hosted on EC2 instances located on subnets contained within a specific VPC. Subnets are associated with a specific availability zone within each AWS region.

Each subnet must be associated with a route table. If no specific route table association has been configured, the subnet will use the default route table that was created when the VPC was first created, called the main route table (discussed next), containing a default route that allows instances within the VPC to communicate with each other. Multiple subnets can be associated and controlled with a single route table that is assigned to multiple subnets. You might have multiple private subnets that need routes to the same service, such as a route to the NAT gateway service enabling resources on private subnets to get updates from a public location on the Internet, or a route to the virtual private gateway (VGW) for VPN connections from external locations.

## The Main Route Table



Each VPC has a default route table called the main route table that provides local routing services throughout each VPC and across all defined availability zones (AZs), as shown in [Figure 4-4](#). The main route table is associated with a VPC after it is first created. The main route table also defines the routing for all subnets that are not explicitly associated with any other custom

route table. The main route table cannot be deleted; however a custom route table can be associated with a subnet, replacing main route table association.

Name	Route Table	Explicit sub	Edge e	Main	VPC ID	Owner
Private_NAT_Access	rtb-00062e5...	subnet-0e...	-	No	vpc-0...	313858614000
	rtb-0da793b...	-	-	Yes	vpc-0...	313858614000
	rtb-0de0c4e...	-	-	Yes	vpc-0...	313858614000
public route dev vpc	rtb-2f124f50	2 subnets	-	No	vpc-6...	313858614000
<b>Default VPC - Main Route Table</b>	<b>rtb-511fb734</b>	-	-	Yes	<b>vpc-c...</b>	<b>313858614000</b>

Route Table: rtb-511fb734

Summary    **Routes**    Subnet Associations    Edge Associations    Route Propagation    Tags

Edit routes

View All routes ▾

Destination	Target	Status
172.30.0.0/16	local	active

**Figure 4-4** Main Route Table

Each custom or default route table has an entry containing the VPC's initial CIDR designations, and a local route used to provide access to the VPC's resources.

As mentioned earlier, you cannot delete the local route entry in a subnet route table, but you can change the local entry to point

to another verified target such as a NAT gateway, network interface, or Gateway Load Balancer endpoint.

## Custom Route Tables

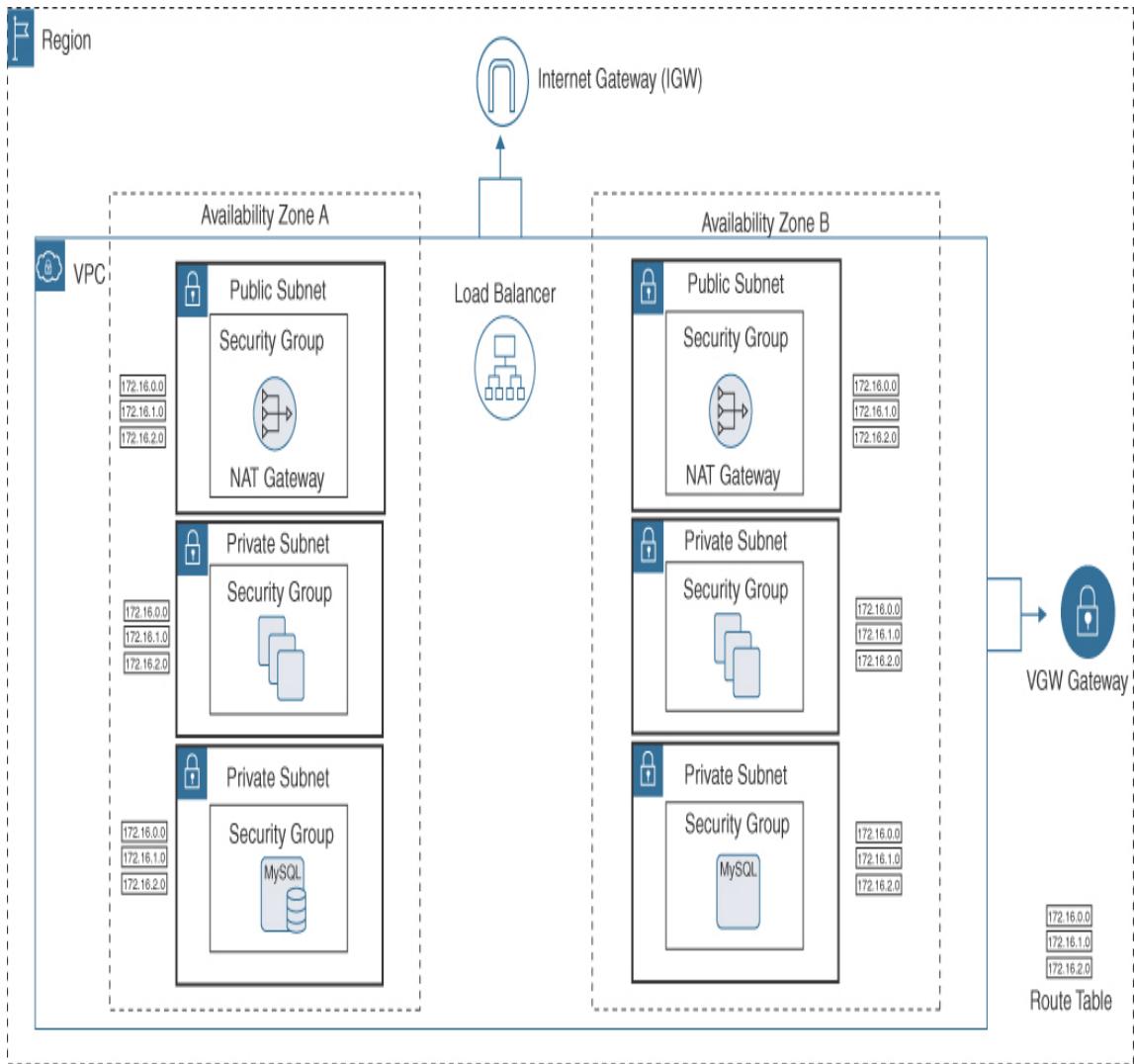


A custom route table is a user-defined routing table that enables custom routes to direct traffic to specific destinations or implement more complex network architectures.

For example, suppose an organization named Terra Firma is considering starting with a two-tier design for the human resources customer relationship management (CRM) application hosted at AWS. For the production workload network, Terra Firma has decided to use two AZs within the VPC hosting the CRM servers to provide high availability and failover for the application and database servers. The following tasks must be carried out to create the required infrastructure for the CRM workload:

- Create public subnets in each AZ.
- Add Elastic Load Balancing (ELB) and NAT services to the public subnets.

- Create separate private subnets for the EC2 instances hosting the CRM application servers and the Amazon RDS MySQL database servers (see [Figure 4-5](#)).



**Figure 4-5** Proposed Two-Tier VPC Subnet Design

The RDS database servers use synchronous replication to ensure database records remain up to date. When synchronous

replication is enabled for an RDS instance, the primary and standby instances are continuously connected, and all data changes made to the primary instance are immediately replicated to the standby instance. This ensures that the standby instance is always up to date and can be quickly switched over to if the primary instance fails.

For the initial infrastructure design, after the subnets have been created, Terra Firma's network administrators must create custom route tables for the following subnet groupings (see [Figure 4-6](#)):

- **Public subnets and custom route tables:** Public subnets host the AWS ELB load balancer and the AWS NAT gateway service. A custom route table will be created and associated with the public subnets, adding a route table entry for the AWS Internet Gateway service. Internet gateway routes are usually set with a destination route of 0.0.0.0/0 as client queries will typically come from multiple source locations across the public Internet.
- **Private subnets and custom route tables:** The application servers are hosted on private subnets within each AZ. The primary and standby database instances will be deployed and managed using the Amazon Relational Database Service (RDS). Separate AWS NAT gateway services with associated

Elastic IP addresses will be ordered and attached to the public subnets in each AZ, enabling the application servers hosted on private subnets to connect to the NAT gateway service and receive any required updates from the Internet. Custom route tables and route table entries pointing to the NAT gateway service must be defined in each private subnet's route table.

Destination	Target	Status	Propagated
192.168.0.0/16	local	active	No
0.0.0.0/0	lgw-021f416882097e0fd	active	No

Add route

Public route table entry for access to the Internet Gateway

Destination	Target	Status	Propagated
192.168.0.0/16	local	active	No
0.0.0.0/0	nat-0948330284faa159b	active	No

Add route

Private route table entry for database instances to access the NAT Gateway service

**Figure 4-6 Using Custom Route Tables**

---

### Note

A single route table can be assigned to multiple subnets within the same VPC.

---

**Key  
Topic**

## Route Table Cheat Sheet

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of route tables:

- Each VPC has a main route table that provides local routing throughout each VPC.
- Each subnet, when created using the VPC dashboard, is implicitly associated with the main route table.
- Don't add additional routes to a main route table. Leaving the main route table in its default state ensures that if the main route table remains associated to a subnet by mistake, the worst that can happen is that local routing is enabled. If additional routes are added to the main route table, the additional routes will be available from each new subnet due to the default association with the main route table.
- The main route table cannot be deleted; however, it can be ignored and will remain unassigned if you do not associate it with any subnets within the VPC.
- Create and assign a custom route table for custom routes required by a subnet.

- Subnet destinations are matched with the most definitive route within the route table that matches the traffic request.

## Security Groups



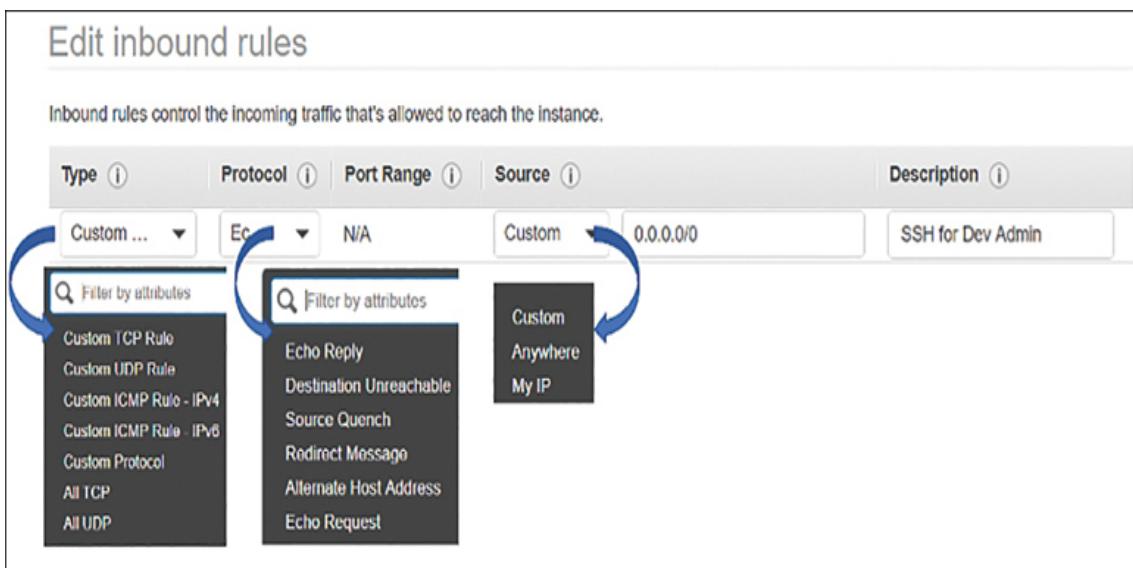
A **security group (SG)** is a virtual software firewall that controls the incoming and outgoing network traffic for one or more EC2 instances hosted in a VPC. Security groups enable you to specify the protocols, ports, and source IP ranges that are allowed to reach your instances. Every attached elastic network interface (ENI) is protected by a security group. Each security group is associated with a specific VPC and has a set of inbound and outbound rules that designate the port(s) and protocol(s) allowed into and out of each network interface, as shown in [Figure 4-7](#).

---

### Note

It might not be obvious when you start creating VPC networking components but all components created using the VPC console are associated. Each subnet, route table, security group, network

interface, and network ACL is assigned to a specific VPC during creation.



**Figure 4-7** Security Group Details

After security groups have been assigned to an EC2 instance, changes made to security groups attached to an EC2 instance while the instance is online usually take effect within seconds.

The initial service quota limit for the number of security groups applied to an EC2 instance is five; you can request an increase using the Service Quotas utility. In addition, every custom security group can have up to 50 inbound and 50 outbound IPv4 or IPv6 rules. You can also increase the number of rules per security group.

Think of each security group as a reusable security template assigned to a particular VPC. Once a security group has been created, it can be assigned multiple times within the VPC where it was created to protect one or many EC2 instances.

One important concept to grasp about security groups is that they don't *deny* traffic flow. Instead, their job is to *allow* traffic flow. Another equally important concept is the direction of traffic flow both inbound and outbound each security group allows.

Security groups are defined as stateful, which means that if traffic is allowed in one direction, the security group automatically allows the traffic in the opposite direction. For example, if you allow incoming traffic on port 80 (HTTP), the security group will automatically allow outgoing traffic on port 80.

A defined inbound port request does not usually use the same port number for the outbound response. For example, if there's a rule defined allowing inbound HTTP traffic across port 80, the outbound traffic response is allowed out; however, the outbound traffic will not use port 80 for outbound communication. Outbound traffic uses a dynamically assigned port called an *ephemeral port*, determined by the operating

system of the server making the response (you will learn more about ephemeral ports later in this chapter, in the upcoming section, “[Network ACLs](#)”).

When a VPC is first created, a default security group is also created. Note that the default security group allows outbound traffic, but any inbound traffic is implicitly denied as inbound rules have not been defined (see [Figure 4-8](#)).

The screenshot shows the AWS Management Console interface for managing security group rules. It is divided into two main sections: 'Inbound rules' and 'Outbound rules'.

**Inbound rules:** This section has a heading 'Inbound rules' with a 'Info' link. Below it, a message states 'This security group has no inbound rules.' A large 'Add rule' button is located at the bottom of this section.

**Outbound rules:** This section has a heading 'Outbound rules' with a 'Info' link. It includes columns for 'Type' (with a 'Info' link), 'Protocol' (with a 'Info' link), 'Port range' (with a 'Info' link), 'Destination' (with a 'Info' link), and 'Description - optional' (with a 'Info' link). There is a search bar and a delete button labeled 'Delete'.

Type	Protocol	Port range	Destination	Description - optional
All traffic	All	All	Anywhere	0.0.0.0/0 X

**Figure 4-8** Default Security Group Rules

The default security group allows all outbound traffic but denies all inbound traffic. EC2 instances in a VPC associated with just the default security group can initiate outbound communications, but all inbound communication is blocked.

The default security groups can't be deleted; however, the default security groups can be removed and custom security groups can be created and used instead.

Here are some additional details about security groups to know:

- A security group is always associated with a single VPC.
- Each elastic network interface assigned to an EC2 instance hosted within a VPC can be associated with up to five security groups by default.
- Security groups allow traffic; however, a security group cannot explicitly deny traffic. If either inbound or outbound access is not specifically allowed, it is implicitly denied.
- When a new security group is created, all outbound traffic is allowed if you don't review and change the default outbound rules before saving.
- Specify which protocols, ports, IP ranges, and security groups are allowed to access your instances.
- Outbound rules can be changed to direct outbound traffic to a specific outbound destination. For example, you could decree that all outbound traffic from a public-facing load balancer can only flow to a security group protecting the web tier.

Security group rules are defined as *stateful*. This means that inbound traffic flowing through a security group is tracked, logging the traffic allowed in, and any allowed inbound traffic is always allowed outbound. This process is called *connection tracking*; connections to a security group are automatically tracked to ensure valid replies. Some AWS services that can be associated with a security group include

- **Amazon Elastic Compute Cloud (EC2)**: Security groups can be used to control network traffic to and from EC2 instances.
- **Amazon Elastic Kubernetes Service (EKS)**: Security groups can be used to control network traffic to and from EKS clusters.
- **Amazon Elastic Container Service (ECS)**: Security groups can be used to control network traffic to and from ECS tasks and services.
- **Amazon Relational Database Service (RDS)**: Security groups can be used to control network traffic to and from RDS instances.
- **Amazon Elastic Load Balancer (ELB)**: Security groups can be used to control network traffic to and from ELB load balancers.

## Security Groups Cheat Sheet

**Key  
Topic**

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of security groups:

- A security group acts like a firewall at the EC2 instance, protecting all attached network interfaces.
- Security groups support both IPv4 and IPv6 traffic.
- A security group controls both outgoing (egress) and incoming (ingress) traffic.
- For each security group, rules control the inbound traffic that is allowed to reach the associated EC2 instances.
- Separate sets of rules control both the inbound and the outbound traffic.
- Each security group includes an outbound rule that allows all outbound traffic by default. Outbound rules can be modified and, if necessary, deleted.
- Security groups allow traffic based on protocols and port numbers.
- Security groups define allow rules. (It is not possible to create rules that explicitly deny access.)
- Security group rules allow you to direct traffic outbound from one security group inbound to another security group

within the same VPC.

- Changes made to a security group take effect immediately.
- Security groups don't deny traffic explicitly; instead, they deny traffic implicitly by defining only *allowed* traffic.
- Security groups are stateful; for requests that are allowed in, their response traffic is allowed out, and vice versa.
- For each rule, you define the protocol, the port or port range, and the source inbound and output destination for the traffic.
- The protocols allowed with security groups are TCP, UDP, or ICMP.

---

#### Note

It is impossible to block specific IP addresses by using a security group; instead, use a network access control list to block a range of IP addresses. Further details are provided in the section [“Network ACLs”](#) later in this chapter.

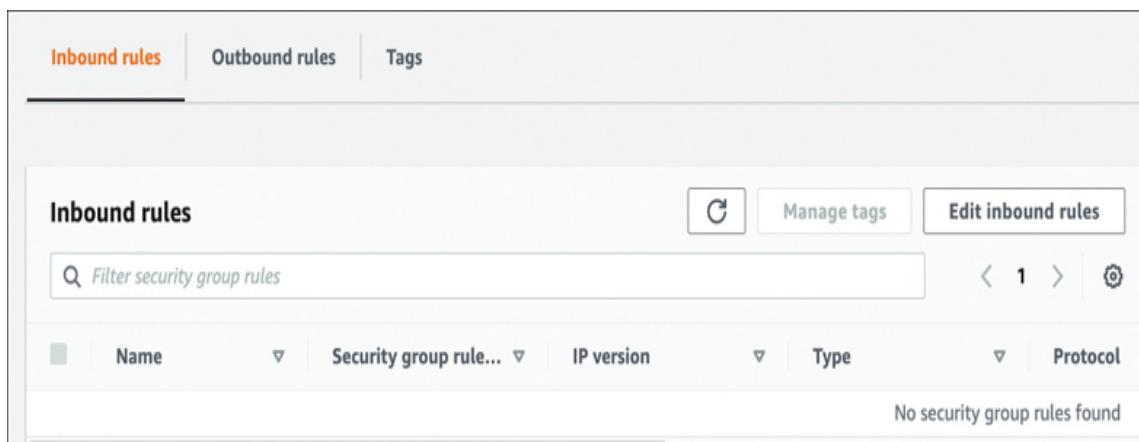
---

## Custom Security Groups

When a custom security group is created, it is associated with a specific VPC. By default, a custom security group allows no inbound traffic but allows all outbound traffic. To create inbound rules and outbound rules, select the security group

properties and use the following tabs to make changes (see [Figure 4-9](#)):

- **Inbound rules:** Define the source of the traffic—that is, where it is coming from—and what the destination port or port range is. The traffic source could be a single IP address (IPv4 or IPv6), a range of addresses, or from another security group.
- **Outbound rules:** Define the destination of the outbound traffic. The destination of the traffic could be a single IP address (IPv4 or IPv6), a range of addresses, or from another security group; EC2 instances that are associated with one security group can access EC2 instances associated with another security group.



**Figure 4-9** Default Security Group Inbound and Outbound Tabs

When designing security groups, the best practice is to minimize the ports allowed. Open only the specific ports that are needed for the services and applications running on your load balancer. The following sections provide some examples to consider for your security group configurations and setup.

---

### Note

Security groups “allow” access; however, security groups can also be said to “deny by default.” If an incoming port is not allowed, access is denied.

---

## Web Server Inbound Ports

Web server security group rules need to allow inbound HTTP or HTTPS traffic access from the Internet (see [Table 4-2](#)). Other rules and inbound ports can also be allowed, depending on workload requirements.

**Table 4-2** Web Server Security Group Options

### Web Server Inbound Ports

Port	Details
80	HTTP port
443	HTTPS port
22	SSH port
3389	Remote Desktop port
445	SMB port
5900	VNC port
8080	HTTP port for port forwarding
4443	HTTPS port for port forwarding

## Web Server Inbound Ports

80 (HTTP)      Inbound IPv4 (0.0.0.0)  
                  Inbound IPv6 (::0)

443              HTTPS

25              SMTP

53              DNS

22              SSH

## Database Server Inbound Ports

Several database server engines are available when deploying Amazon Relational Database Server database instances. The default port address assigned during deployment is based on the database engine's default port, which can be changed to a custom port number for additional security. [Table 4-3](#) lists the default RDS security group database port options that are assigned per database engine during installation.

**Table 4-3** RDS Database Inbound Ports

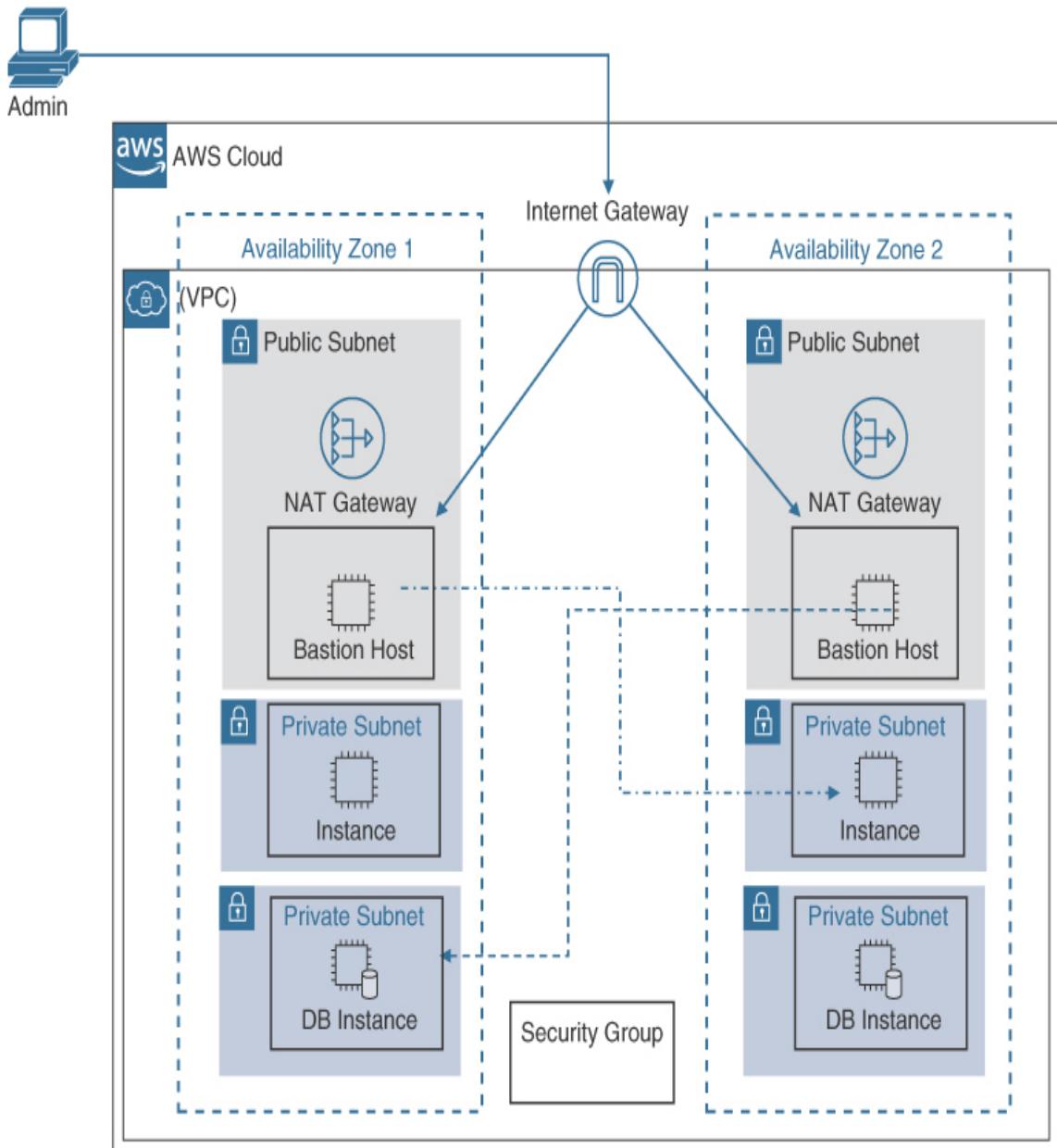
Port	Database Engine
3306	Microsoft SQL Server
3306	Amazon Aurora/MySQL
5432	Amazon Aurora PostgreSQL
1521	Oracle
27017	MongoDB

## Administration Access

Connecting to an EC2 instance to perform direct administration requires associating a security group with an elastic network interface with inbound rules allowing either Secure Shell (SSH) or Remote Desktop Protocol (RDP) access, depending on the host's operating system (see [Table 4-4](#)). Deploying an EC2 instance as a bastion host on a public subnet would allow administrators to first authenticate to the bastion host and then “jump” to the associated EC2 instance in the private subnet.

A bastion host is a special purpose EC2 instance or third-party software appliance hosted on a public subnet exposed to the Internet, serving as a secure gateway or “jump box” for remote access to instances hosted on private subnets in the VPC without exposing a web or database server directly to the Internet (see [Figure 4-10](#)). Common security group settings for a bastion host could include the following:

- Allow incoming traffic on port 22 for SSH access.
- Allow incoming traffic on port 443 for HTTPS access.
- Allow outgoing traffic to allow the bastion host to access other machines on the network.
- Allow outgoing traffic linking specific security groups to allow the bastion host to access specific machines on the network.
- Set the source IP range for incoming traffic to a restricted range, such as the IP addresses of your office or trusted administrators.



**Figure 4-10** Bastion Host Solution

**Table 4-4** Security Groups Inbound Ports for Administrative Access

Port	Operating System
------	------------------

22	Linux
----	-------

3389	Windows
------	---------

## Understanding Ephemeral Ports

**Key Topic**

When configuring network security settings, such as a security group or network access control list, you need to allow for ephemeral port ranges for outbound communication from your instances or network services, such as load balancers.

Network design needs to consider where the traffic originated for both inbound and outbound traffic requests. Return traffic from an EC2 instance hosted in a VPC to a destination across the Internet communicates using a dynamic outbound or inbound *ephemeral port*.

Ephemeral ports are temporary, short-lived ports that are typically used by client applications for outbound

communications from a predefined range of port numbers and are used for the duration of a communication session. When the session is complete, the port is released and can be used by another application.

TCP/IP communications don't utilize the same inbound and outbound ports; instead, the client or server's operating system defines the range of ports that will be dynamically selected for the return communication—that is, the outbound communication. Network connections require two endpoints: a source and a destination. Each source and destination endpoint has an IP address and an associated port number.

When a client system connects to a server, several components are employed: the server IP address, the server port, the client IP address, and the client port. The ephemeral port is a temporary port assigned by the computer's TCP/IP stack. The TCP/IP implementation chooses the port number based on the host operating system. In the case of Windows Server 2016 and above, the ephemeral port range is from 49152 to 65535. If Linux is the operating system, the ephemeral port range is from 32768 to 61000, as shown in [Table 4-5](#). Different operating system versions may use slightly different ephemeral ranges; check what your operating system uses for ephemeral ports.

When communication is carried out from a source service to its destination, the traffic typically uses the named port for the destination traffic, such as port 22 on a Linux box accepting SSH connections. However, for the return traffic from the server to the client, an ephemeral port is typically used for the return traffic. An ephemeral port can be defined as a dynamically assigned port from a range of assumed available port addresses. Outbound packets travel through an outbound port allowed by the existing security group using an allowed ephemeral port.

Outbound communication from an EC2 instance hosted on a VPC must have an allowed outbound range of ephemeral ports. These ports remain available only during the communication session; each dynamically assigned port is released after the TCP connection terminates. If custom security groups or NACLs are deployed, ephemeral rules need to appear in both the inbound and outbound rules to cover the dynamic requirements of communication using ephemeral ports. [Table 4-5](#) lists some common inbound port numbers that are typically used. The outbound port 443 is the exception as it answers outbound, using port 443.

**Table 4-5** Inbound Port Numbers

---

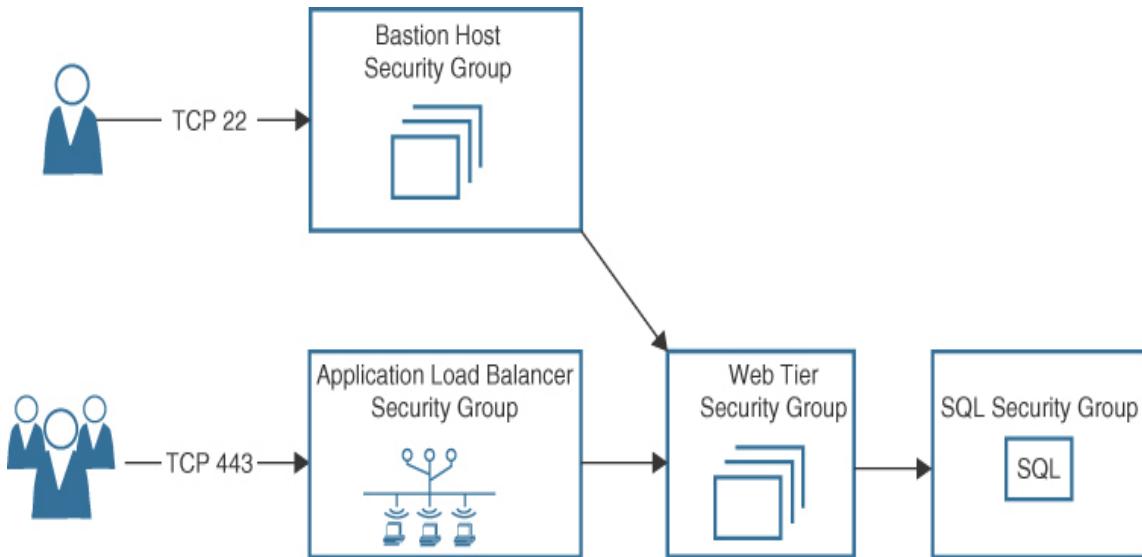
Port #	Service	Protocol	Description	Port Type
20	FTP	TCP/UDP	File transfer data	Dynamic
21	FTP	TCP/UDP	File transfer control	Dynamic
22	SSH	TCP/UDP/SCTP	Secure Shell	Dynamic
25	SMTP	TCP/UDP	Simple mail transfer	Dynamic
67	BOOTPS	UDP	Bootstrap (BOOTP/DHCP) server	Dynamic
68	BOOTPC	UDP	Bootstrap (BOOTP/DHCP) client	Dynamic

Port #	Service	Protocol	Description	Port Type
69	TFTP	UDP	Trivial file transfer	Dynamic
80	HTTP	TCP	Hypertext Transfer Protocol	Dynamic
88	Kerberos	TCP	Kerberos	Dynamic
123	NTP	UDP	Network time	Dynamic
443	HTTPS	TCP	HTTP over TLS/SSL	443
143	Microsoft-ds	IMAP	Internet Message Access Protocol	Dynamic

## Security Group Planning

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of security group design:

- Create a security group for your public-facing application load balancer that accepts inbound traffic from the Internet (port 80 or port 443) and sends outbound traffic to your web tier security group, as shown in [Figure 4-11](#).
- Create separate security groups for administrative tasks.
- Create a security group for your application tier that only accepts inbound traffic from the web tier and sends outbound traffic to your database tier security group.
- Create a security group for your database tier that only accepts inbound traffic from the application tier.
- Deploy a test application and test communication on a test VPC before deploying to production.



**Figure 4-11** Security Group Design

## Network ACLs

### Key Topic

A ***network access control list (NACL)*** is an optional software firewall that controls inbound and outbound traffic for each subnet within a VPC. A NACL is a set of rules that allows or denies traffic based on the source and destination IP addresses, ports, and protocols. Both the Transmission Control Protocol (TCP) and User Datagram Protocol (UDP) are supported by network ACLs.

NACLs are used to supplement the security provided by security groups, which are associated with individual Amazon EC2 instances. Whereas security groups control traffic to and from specific instances, network ACLs control traffic at the subnet level, allowing you to set rules for the subnet.

Each VPC is associated with a default NACL that is merely a placeholder as the default network ACL allows all inbound and outbound traffic at the subnet level. Custom NACLs can and should be created, just like security groups. NACLs, once created, can also be associated with one or multiple subnets within the associated VPC.

Each NACL contains a set of inbound and outbound subnet traffic rules, from a starting lowest-numbered rule to the highest-numbered rule, as shown in [Figure 4-12](#). Rules are processed in order to determine whether traffic is allowed or denied inbound or outbound on each subnet.

The diagram illustrates two Network ACL (NACL) tables. The top table, labeled 'Inbound NACL', has columns for Rule#, Source IP, Protocol, Port, Allow / Deny, and Comments. It contains four rules: rule 100 allows TCP port 22 from a private IP range; rule 110 allows TCP port 3389 from a private IP range; rule 120 allows TCP ports 32768-65535 from a private IP range; and a catch-all rule '\*' denies all traffic from 0.0.0.0/0. The bottom table, labeled 'Outbound NACL', also has columns for Rule#, Source IP, Protocol, Port, Allow / Deny, and Comments. It contains three rules: rule 100 allows all traffic from a private IP range to all destinations; rule 120 allows TCP port 32768-65535 from a private IP range to a private network; and a catch-all rule '\*' denies all traffic from 0.0.0.0/0. Annotations with arrows point from the table rows to their respective comments: rule 120's comment 'Inbound return traffic to subnet' points to rule 120 in the Inbound NACL, and rule '\*'s comment 'Denies inbound traffic not handled by existing rule' points to rule '\*' in the Inbound NACL. Another annotation 'Inbound return traffic from private subnet requests' points to rule '\*' in the Outbound NACL. A final annotation 'Outbound responses to clients on private network' points to rule 100 in the Outbound NACL.

Rule#	Source IP	Protocol	Port	Allow / Deny	Comments
100	Private IP address range	TCP	22	ALLOW	Inbound SSH to subnet
110	Private IP address range	TCP	3389	ALLOW	Inbound SSH to subnet
120	Private IP address range	TCP	32768-65535	ALLOW	Inbound return traffic to subnet
*	0.0.0.0/0	All	All	DENY	Denies inbound traffic not handled by existing rule

Rule#	Source IP	Protocol	Port	Allow / Deny	Comments
100	Private IP address range	All	All	ALLOW	Outbound traffic to private network
120	Private IP address range	TCP	32768-65535	ALLOW	Outbound traffic to private network
*	0.0.0.0/0	All	All	DENY	Denies outbound traffic not handled by existing rule

**Figure 4-12** NACL Design

NACLs are located at the perimeter of each subnet and provide an additional layer of defense. A single NACL can protect multiple application servers at the subnet level. Rules can target an entire subnet or a block of IP addresses.

---

### Note

Security protection provided by a NACL is at the subnet level. Blocked network traffic denied at the subnet level cannot get anywhere near your EC2 instances.

---

## Network ACL Implementation Details

Both inbound and outbound rules should be numbered in an organized fashion with some separation between the numbers so that you can make changes if necessary in the future. It is best practice to number your inbound and outbound rules by 10s—10 for the first rule, 20 for the second rule, and so on.

NACL rules are *stateless*; this means that inbound and outbound NACL rules are independent from each other.

- Outbound rules are processed separately without any regard to the defined inbound rules.
- Inbound rules are processed without any regard to the outbound rules that have been defined.

## Network ACL Cheat Sheet



For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of NACLs:

- A NACL is an optional security control for subnets.
- Each VPC is assigned a default NACL that allows all inbound and outbound traffic across all subnets.

- NACLs are stateless in design; inbound and outbound rules are enforced independently.
- Each NACL is a collection of deny or allow rules for both inbound and outbound traffic.
- The default NACL can be modified.
- A NACL has both allow and deny rules.
- A NACL applies to both ingress and egress subnet traffic; it does not apply to traffic within the subnet.
- You can create custom NACLs and associate them to any subnet in a VPC.
- A custom NACL can be associated with more than one subnet.
- A subnet can be associated with only one NACL.
- A NACL is a first line of defense at the subnet level; a security group is a second line of defense at the instance.

## Network ACL Rule Processing

Both inbound and outbound rules are evaluated, starting with the lowest-numbered defined rule. Once a rule matches the traffic request, it is applied; there is no additional comparison with higher-numbered rules that may also match. A misconfigured lower-numbered rule that also matches the same traffic request could cause problems. If you designated a higher-numbered rule to deal with specific traffic, but instead a

lower-numbered rule matched the traffic request, the higher-numbered rule would never be used, as shown in [Table 4-6](#).

**Table 4-6** NACL Rules with Incorrect Order

Rule Number	Source	Protocol	Port Number	Allow/Deny
100	0.0.0.0/0	TCP	22	Allow
110	0.0.0.0/0	TCP	3389	Allow
120	0.0.0.0/0	TCP	3389	Deny

Rule Number	Source	Protocol	Port Number	Allow/Deny
*	0.0.0.0/0	All	All	Deny



\* All undefined traffic is blocked.

When inbound packets appear at the subnet level, they are evaluated against the incoming (ingress) rules of the network ACL. For example, the request is for port 443. Starting with the first rule, numbered 100, there is not a match because the first rule has been defined for port 80 HTML traffic (see [Table 4-7](#)). The second rule, numbered 110, has been defined for allowing HTTPS traffic. Therefore, HTTP traffic is allowed onto the subnet. All other traffic is denied access if it doesn't match any of the inbound allow rules. If the inbound communication is from the Internet, the source is defined as 0.0.0.0/0 because the traffic could come from any location.

Outbound or egress traffic also must be matched with an outbound rule for the traffic to be allowed to exit the subnet. The outbound rule for HTTPS traffic also uses port 443; the destination is 0.0.0.0/0 because the destination could be anywhere across the Internet. In this case, both the inbound and the outbound rules for HTTPS traffic is set to allow. A rule for the required range of dynamic ports allows outbound responses.

**Table 4-7** Custom NACL Setup

Inbound Network ACL				
Rule	Source Address	Protocol	Port Number	Allow/Deny
100	0.0.0.0/0	TCP	80	Allow

Inbound Network ACL				
110	0.0.0.0/0	TCP	443	Allow

120	IPv4 address range for administration	TCP	22	Allow
-----	---------------------------------------	-----	----	-------

130	IPv4 address range for administration	TCP	3389	Allow
-----	---------------------------------------	-----	------	-------

*	0.0.0.0/0	All	All	Deny
---	-----------	-----	-----	------

## Inbound Network ACL

<b>Rule</b>	<b>Destination</b>	<b>Protocol</b>	<b>Port</b>	<b>Allow/D</b>
	<b>IP Address</b>			
100	0.0.0.0/0	TCP	80	Allow
110	0.0.0.0/0	TCP	443	Allow

Inbound Network ACL				
120	0.0.0.0/0	TCP	32768–	Allow
			65535	
*	0.0.0.0/0	All	All	Deny

\* All undefined traffic is blocked.

## VPC Flow Logs



VPC flow logs enable you to capture information about the IP traffic going to and from a VPC. Flow logs can be used to monitor, troubleshoot, and analyze the network traffic in your VPC.

VPC flow logs can be enabled at the VPC, subnet, or elastic network interface level, capturing traffic flowing in and out of the specified resource. Flow logs record the IP traffic flowing in and out of your VPC, including information about the source and destination IP addresses, ports, protocols, and packet and byte counts.

Network traffic can be captured for analysis or to diagnose communication problems at the level of the elastic network interface, subnet, or entire VPC. When each flow log is created, define the type of traffic that will be captured—accepted traffic, rejected traffic, or all traffic. AWS does not charge for creating a flow log but will impose charges for log data storage.

Flow logs can be stored either as CloudWatch logs or directly in an S3 bucket, as shown in [Figure 4-13](#). If VPC flow logs are stored as CloudWatch logs, AWS IAM roles must be created that define the permissions allowing the CloudWatch monitoring service to publish the flow log data to the CloudWatch log group. Once a log group has been created, you can publish multiple flow logs to the same log group.

## Flow log settings

Name - optional

Filter  
The type of traffic to capture (accepted traffic only, rejected traffic only, or all traffic).  
 Accept  
 Reject  
 All

Maximum aggregation interval [Info](#)  
The maximum interval of time during which a flow of packets is captured and aggregated into a flow log record.  
 10 minutes  
 1 minute

Destination  
The destination to which to publish the flow log data.  
 Send to CloudWatch Logs  
 Send to an Amazon S3 bucket

Destination log group [Info](#)  
The name of the Amazon CloudWatch log group to which the flow log is published. A new log stream is created for each monitored network interface.  
 ▼ C

**Figure 4-13** Flow Log Storage Location Choices

If you create a flow log for a subnet, or VPC, each network interface in the subnet or VPC is monitored. Launching additional EC2 instances into a subnet with an attached flow log results in new log streams for each new network interface and network traffic flows.

Not all traffic is logged in a flow log. Examples of traffic that is not logged in flow logs include AWS Route 53 server traffic, Windows license activation traffic, EC2 instance metadata requests, Amazon Time Sync Service traffic, reserved IP address traffic, and DHCP traffic.

Any EC2 instance elastic network interface can be tracked with flow logs. Here are several examples of where VPC flow could be useful:

- **Amazon Elastic Compute Cloud (EC2):** VPC flow logs can be enabled for EC2 instances to capture traffic flowing to and from the instances.
- **Amazon Elastic Load Balancer (ELB):** VPC flow logs can be enabled for ELB load balancers to capture traffic flowing to and from the load balancer.
- **Amazon Elastic Kubernetes Service (EKS):** VPC flow logs can be enabled for EKS clusters to capture traffic flowing to and from the cluster.
- **Amazon Elastic Container Service (ECS):** VPC flow logs can be enabled for ECS tasks and services to capture traffic flowing to and from the tasks and services.
- **Amazon Route 53:** VPC flow logs can be enabled for Route 53 to capture traffic flowing to and from Route 53.

## NAT Services



At AWS, the purpose of network address translation (NAT) services is to provide an indirect path for EC2 instances hosted on private subnets that need Internet access to obtain updates, licensing, or other external resources. NAT is a networking technique that enables private network resources to access the Internet while hiding their true IP addresses. Several AWS services provide NAT capabilities:

- **Amazon Virtual Private Cloud (VPC) NAT Gateway:** Enables instances in a private subnet to access the Internet without exposing their private IP addresses.
- **AWS Transit Gateway NAT:** Enables instances in a VPC or on-premises network to access the Internet without exposing their private IP addresses.
- **AWS PrivateLink NAT Gateway:** Enables instances in a VPC to access resources in another VPC or on-premises network without exposing their private IP addresses.

## NAT Gateway Service

Amazon VPC NAT Gateway is a service that provides NAT capabilities for Amazon VPC, allowing instances in private subnets to indirectly access the Internet. The NAT gateway translates the private IP addresses of the EC2 instance requesting access to its own public IP address, allowing EC2 instances hosted on private subnets to access the Internet without exposing their private IP addresses.

The **NAT gateway service** is hosted in a public subnet configured with an Elastic IP address (a static public IP address), as shown in [Figure 4-14](#) for Internet communication. For multi-availability redundancy, Amazon recommends placing a NAT gateway in each availability zone. Route table entries need to be added to each private subnet's route table, allowing EC2 instances with a path to access the NAT gateway service.

**Create NAT gateway** Info

Create a NAT gateway and assign it an Elastic IP address.

**NAT gateway settings**

**Name - optional**  
Create a tag with a key of 'Name' and a value that you specify.

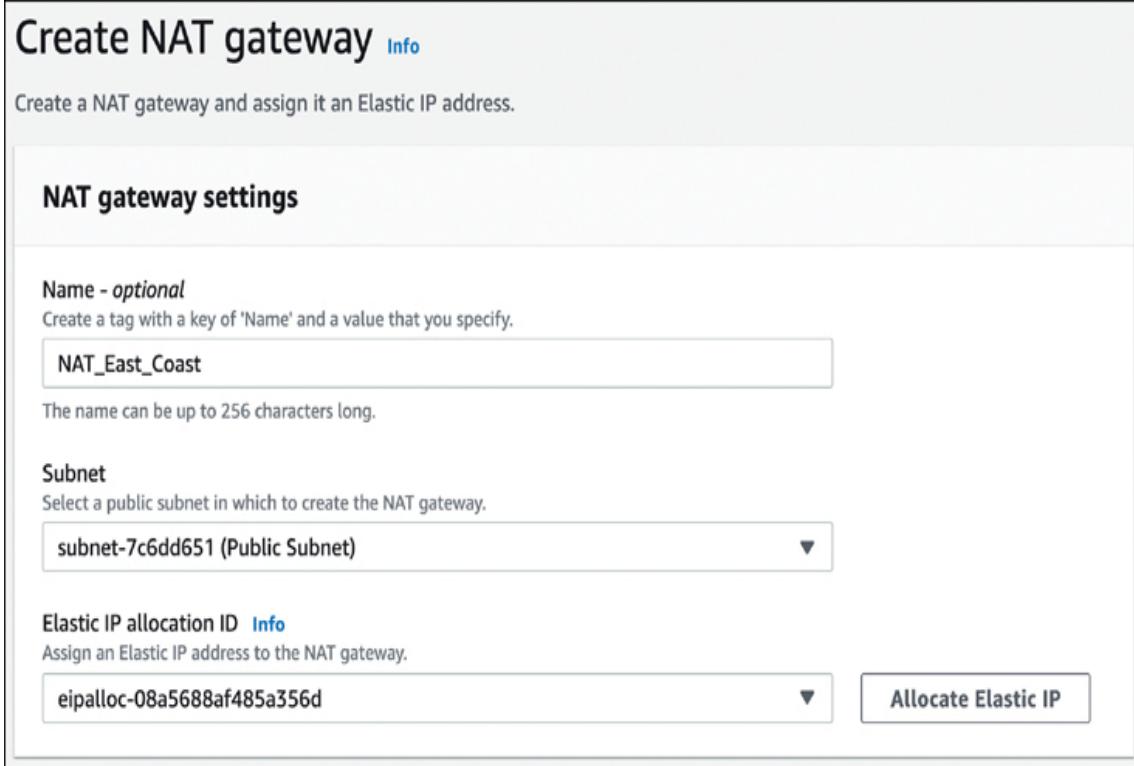
The name can be up to 256 characters long.

**Subnet**  
Select a public subnet in which to create the NAT gateway.

▾

**Elastic IP allocation ID** Info  
Assign an Elastic IP address to the NAT gateway.

▾



**Figure 4-14** Creating a NAT Gateway

---

### Note

The AWS NAT Gateway service initially supports up to 5 Gbps of bandwidth throughput and can scale up to 50 Gbps, as required.

---

### NAT Instance

A NAT gateway third-party software appliance could also be deployed in a public subnet to allow EC2 instances in a private

subnet to connect to the Internet and receive updates as necessary. However, you must configure and manage each NAT instance that is deployed. If you decide to use a third-party solution to provide NAT services, Amazon recommends that you create a high-availability pair of NAT instances for redundancy.

[\*\*Table 4-8\*\*](#) compares the NAT gateway service and the NAT instance.

**Table 4-8** NAT Gateway and NAT EC2 Comparison

Parameter	NAT Gateway Service	NAT Instance
Management	By AWS	By the customer
Bandwidth	Up to 50 Gbps	Depends on the EC2 instance size
Maintenance	By AWS	By the customer
Public IP address	Elastic IP address	Elastic IP address

Security groups	Not supported	Required
-----------------	---------------	----------

Port forwarding	Not supported	Supported
-----------------	---------------	-----------

Bastion host	Not supported	Supported
--------------	---------------	-----------

---

### Note

When deploying a NAT instance, source/destination checks must be disabled on the EC2 instance. By default, source/destination checks are enabled for all EC2 instances, which means that the instance can send and receive traffic only if the source and destination IP addresses match the private IP address of the instance.

---

## AWS NAT Gateway Service Cheat Sheet

**Key Topic**

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of the NAT Gateway service:

- An AWS NAT gateway must be hosted in a public subnet.
- An AWS NAT gateway uses an Elastic IP address as its static public IP address.
- The AWS NAT gateway service does not support security groups.
- The AWS NAT gateway service does not support port forwarding.

## Amazon Cognito



Amazon Cognito provides authentication, authorization, and user management for web and mobile applications. Amazon Cognito enables users to sign into applications hosted at AWS using popular identity providers, such as Amazon, Facebook, and Google, without having to create new credentials. End users sign in using either a user pool or federated identity provider (see [Figure 4-15](#)).

## Configure sign-in experience Info

Your app users can sign in to your user pool with a user name and password, or sign in with a third-party identity provider.

### Authentication providers

Configure the providers that are available to users when they sign in.

#### Provider types

Choose whether users will sign in to your Cognito user pool, a federated identity provider, or both. Amazon Cognito has different pricing for federated users and user pool users. [Learn more about pricing](#)

##### Cognito user pool

Users can sign in using their email address, phone number, or user name. User attributes, group memberships, and security settings will be stored and configured in your user pool.

##### Federated identity providers

Users can sign in using credentials from social identity providers like Facebook, Google, Amazon, and Apple; or using credentials from external directories through SAML or Open ID Connect. You can manage user attribute mappings and security for federated users in your user pool.

### Cognito user pool sign-in options Info

Choose the attributes in your user pool that are used to sign in. If you select only one attribute, or you select a user name and at least one other attribute, your user can sign in with all of the selected options. If you select only phone number and email, your user will be prompted to select one of the two sign-in options when they sign up.

User name

Email

Phone number



Cognito user pool sign-in options can't be changed after the user pool has been created.

**Figure 4-15** AWS Cognito Authentication Options

## User Pool

Amazon Cognito user pools are a fully managed user directory that enables you to create and manage user accounts for your

application. User pools provide sign-up and sign-in options for your users, as well as user profile management and security features such as multi-factor authentication and password policies.

A member of a user pool can sign into a web application with a username, phone number, or email address. Multi-factor authentication (see [Figure 4-16](#)) is supported during the sign-in process using an authenticator app such as Authy or Google Authenticator or an SMS message for the time-based one-time password (TOTP).

## Attribute verification and user account confirmation

Choose between Cognito-assisted and self-managed user attribute verification and account confirmation. Only verified attributes can be used for sign-in, account recovery, and MFA. A user account must be confirmed either by attribute verification, or user pool administrator confirmation, before a user is allowed to sign in.

### Cognito-assisted verification and confirmation [Info](#)

#### Allow Cognito to automatically send messages to verify and confirm - Recommended

Cognito sends a verification message with a code that the user must enter. For new users, this will verify the attribute and confirm their account. When this feature is not enabled, administrative API operations and Lambda triggers verify and confirm users.

#### Attributes to verify [Info](#)

Choose the user contact attribute that Cognito will send a verification message to. Recipient message and data rates apply when you use SMS.

##### Send SMS message, verify phone number

Verify with SMS to allow users to use their phone number for sign-in, MFA, and account recovery. SMS messages are charged separately by Amazon SNS.

##### Send email message, verify email address

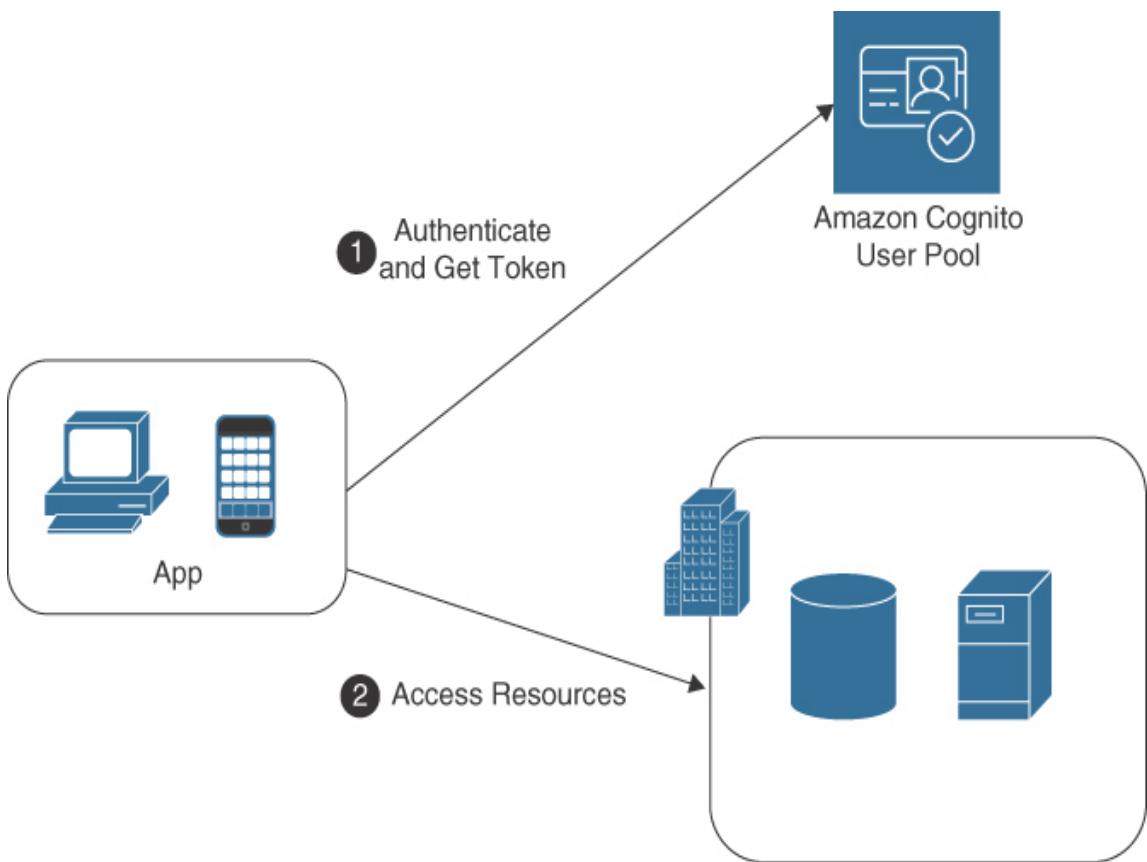
Verify with email to allow users to use their email address for sign-in, MFA, and account recovery. Email messages are charged separately by Amazon SES.

##### Send SMS message if phone number is available, otherwise send email message

You must build custom code when you want to verify both email and phone numbers at user account creation.

**Figure 4-16** Multi-Factor Authentication Options

After an end user has been successfully authenticated using Amazon Cognito, a JSON Web Token (JWT) is issued to secure API communications or to be exchanged for temporary credentials allowing access to on-premises resources or AWS resources such as the S3 storage services used by the web or mobile application (see [Figure 4-17](#)).

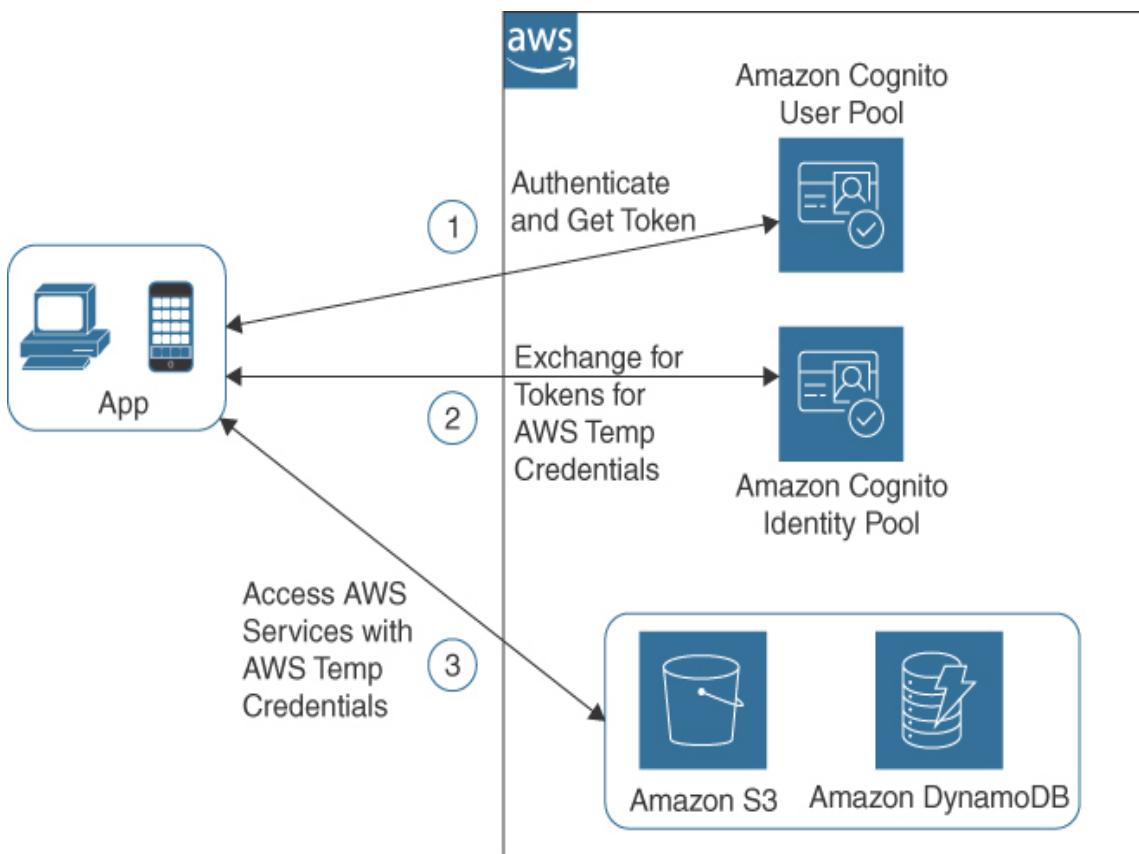


**Figure 4-17** User Pool Sign-in

## Federated Identity Provider

Users can authenticate to a web or mobile app using a social identity provider such as Google, Facebook, or Apple, or using a Security Association Markup Language (SAML) provider such as Active Directory Federation Services or OpenID Connect (OIDC). After a successful user pool authentication, the user pool tokens are forwarded to the AWS Cognito identity pool, which provides temporary access to AWS services (see [Figure 4-18](#)). Amazon Cognito identity pools enable you to grant your users access to

AWS services, such as Amazon S3 and Amazon DynamoDB. Identity pools enable your users to sign in to your application and use AWS resources without having to create AWS credentials.



**Figure 4-18** User Pool and Federated Identity Pool

---

### Note

The AWS Amplify framework can be used to create an end-user application that integrates with

Amazon Cognito.

---

## External Connections



Many companies design solutions using a private hybrid design, where the corporate data center is securely connected to AWS using an AWS VPN connection. Using an IPsec VPN connection to connect to your VPC provides a high level of security.

Before a VPN connection can be set up and connected from your corporate data center to a VPC from your remote network, you need a *virtual private gateway (VPG)* directly attached to the VPC where access is required.

---

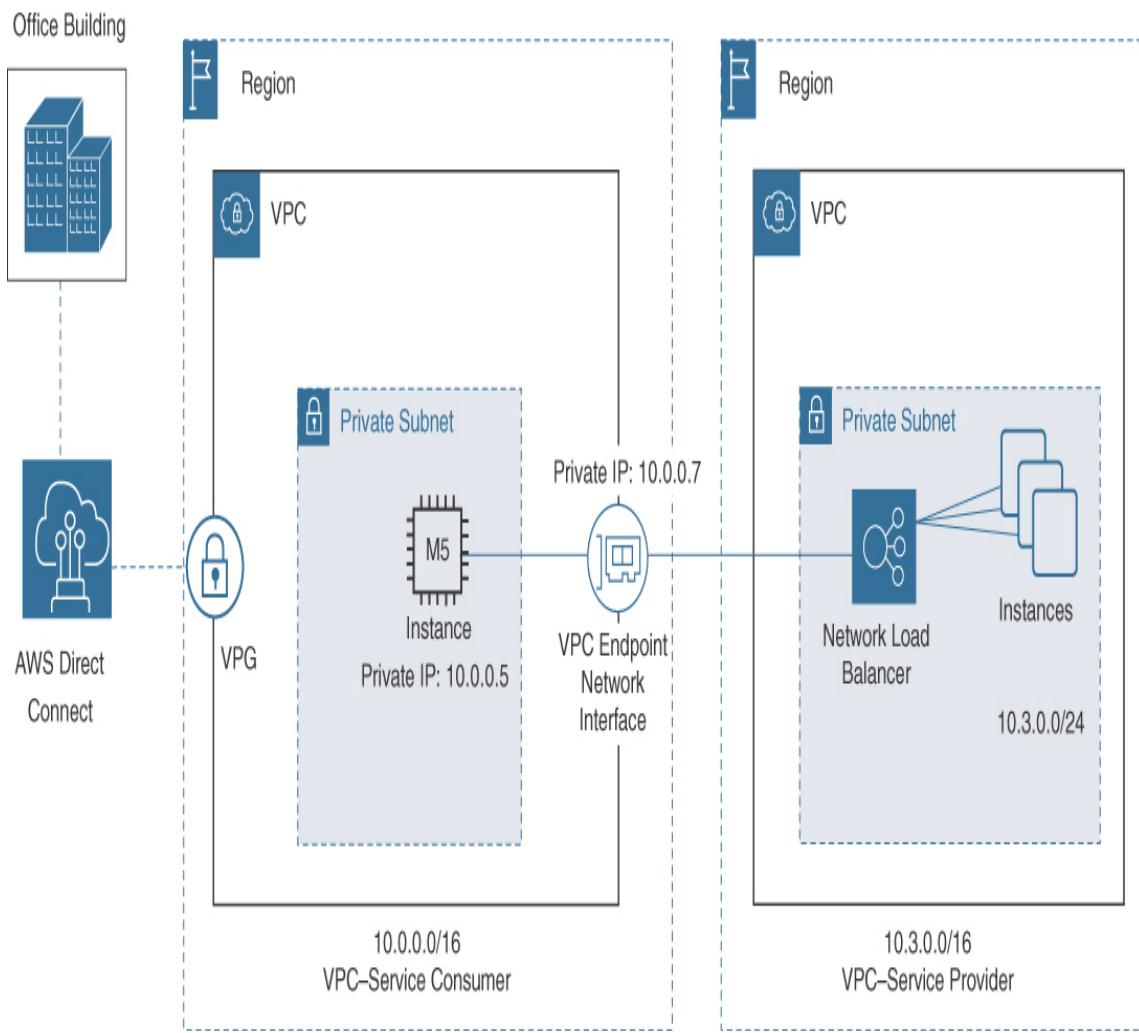
### Note

Both an Internet gateway (IGW) and a VPG are directly attached to the VPC and not subnets.

Gateway devices require route table entries on each subnet where access is required.

---

Routing types supported are either static or dynamic routes using Border Gateway Protocol (BGP). A VPN connection with a single static route is shown in [Figure 4-19](#).



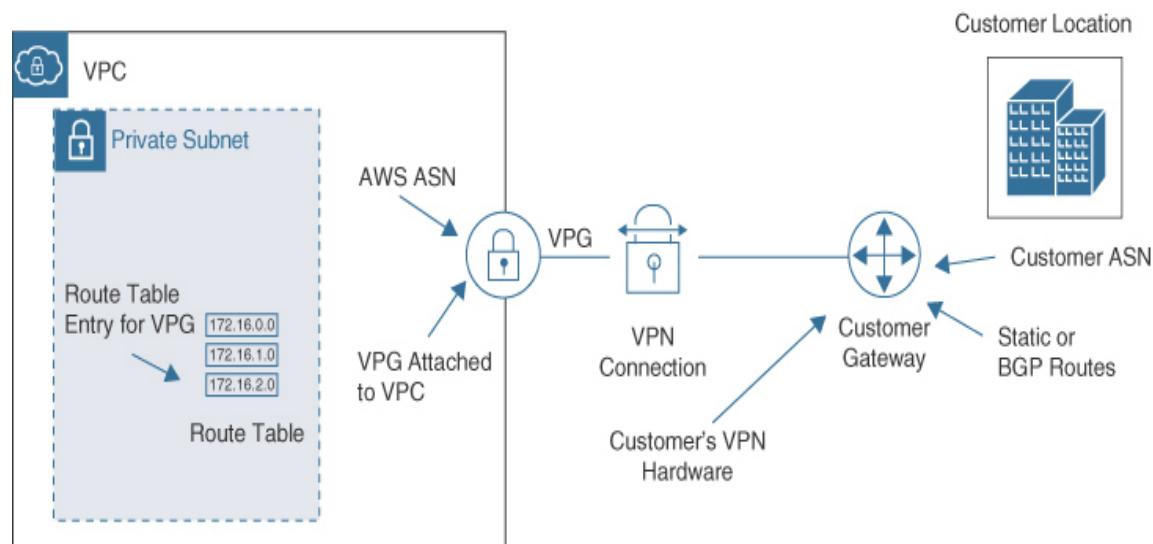
**Figure 4-19** External Private Connection Choices

Each VPN connection at AWS is created with two endpoints; each endpoint is connected to a separate availability zone and assigned a unique public IP address.

## Virtual Private Gateway

A virtual private gateway is the VPN concentrator on the AWS VPC. The virtual private gateway uses Internet Protocol Security (IPsec) to encrypt the data transmitted between the on-premises network and the VPC. When creating a site-to-site VPN connection, create a virtual private gateway on the AWS side of the connection and a customer gateway on the customer side of the connection.

Several AWS components are required to be set up and configured for an AWS VPN connection. [Figure 4-20](#) shows the common components: the VPG, the customer gateway (CGW), and the VPN connection.



**Figure 4-20** VPG Connection Components Choices

## **Customer Gateway**

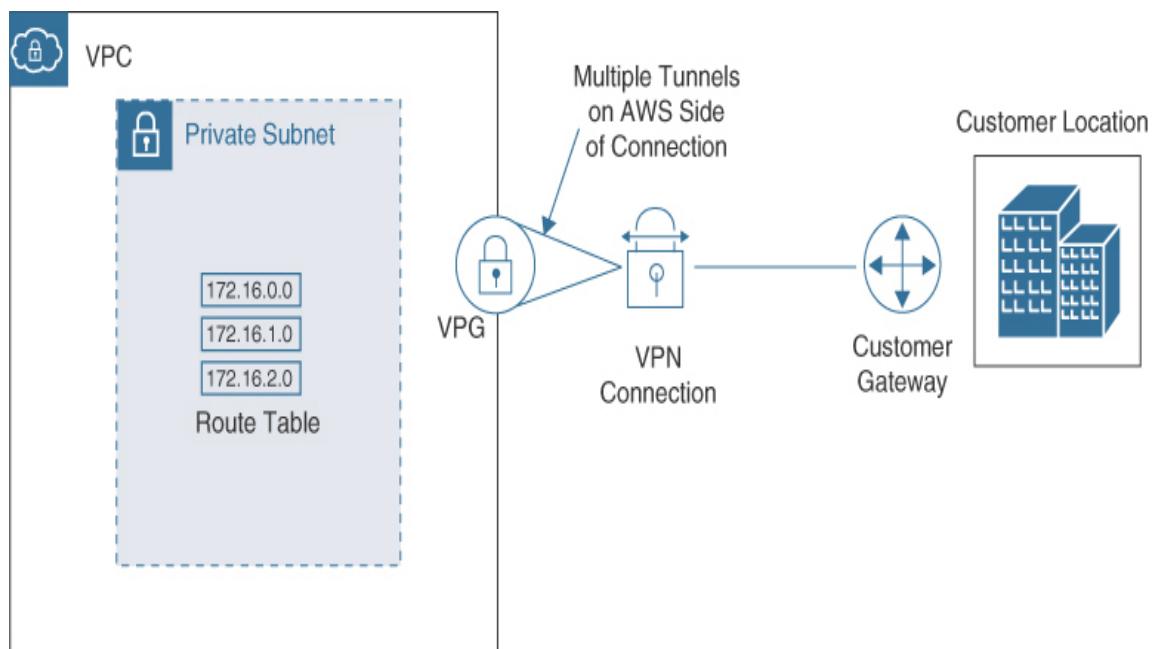
The customer gateway is the VPN concentrator on the customer side of a site-to-site VPN connection.

The customer gateway provides the VPN endpoint for your on-premises network and uses IPsec to encrypt the data transmitted between the on-premises network and the VPC. The customer gateway device provided must be compatible with AWS VPN connections. Customers use hardware or virtual devices for their customer gateway devices. AWS provides configuration steps for most of the popular customer hardware options. Examples of devices that AWS supports include Cisco, Check Point, Fortinet, Juniper, and Palo Alto.

During installation, you will be prompted to download the configuration file that matches your customer gateway device. Information contained in this document includes device details and tunnel configuration.

When creating a customer gateway, enter the public IP address or the private certificate of your customer gateway device and indicate the type of routing to be used: static or dynamic. If you choose dynamic routing, enter your private autonomous system number (ASN) for border gateway protocol (BGP) communications. When connections are completed on both the

customer and AWS sides, traffic requests from the customer side of the AWS VPN connection initiate the VPN tunnel, as shown in [Figure 4-21](#).



**Figure 4-21** AWS VPN Tunnel Connections Choices

---

#### Note

During the configuration of an AWS VPN connection, you can accept the ASN provided by AWS or specify your custom ASN number.

---

Some of the most common routing options for AWS VPN connections include

- **Static routing:** Static routing enables you to specify routes for traffic over a VPN connection. With static routing, specify the IP address ranges and destinations for your traffic, and the VPN connection will use this information to route traffic.
- **Dynamic routing:** With dynamic routing, the AWS VPN connection will automatically add and remove routes as needed, based on the traffic paths available.

## AWS Managed VPN Connection Options

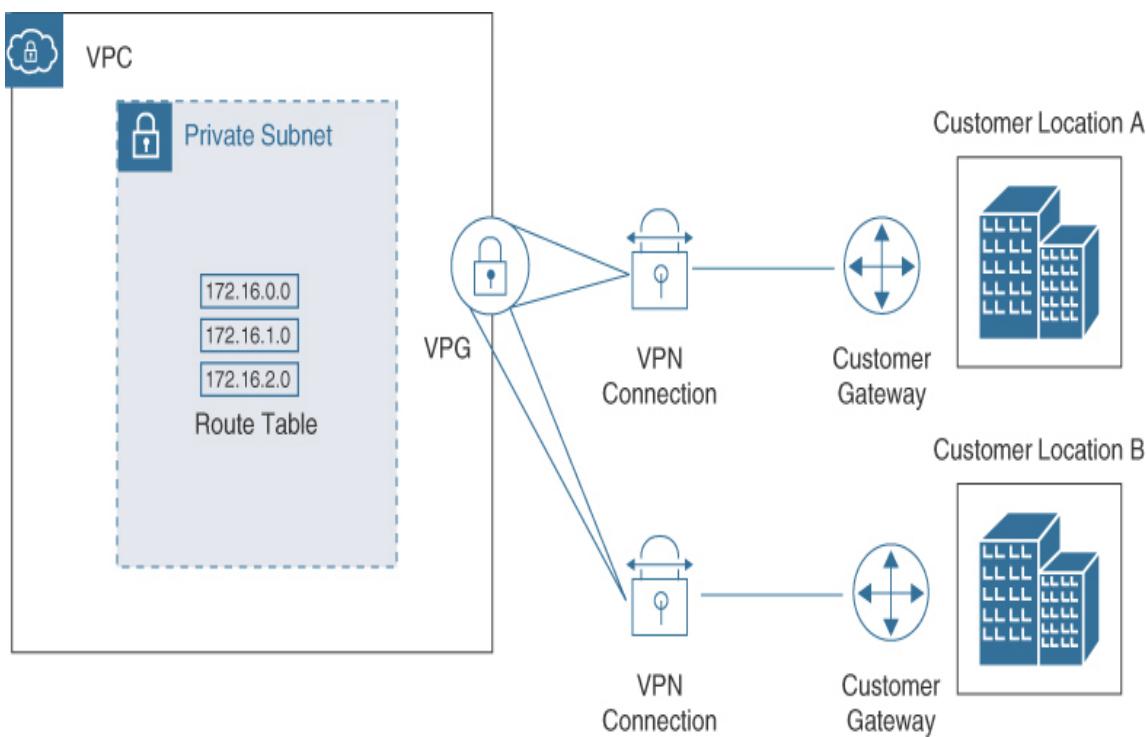
Common solutions for AWS VPN connections include

- **AWS Site-to-Site VPN:** Enables you to create a secure connection between your on-premises network and your VPC. This type of VPN connection uses IPsec to encrypt the data transmitted between the on-premises network and the VPC.
- **AWS Client VPN:** Enables you to create a secure, encrypted connection between your VPC and your remote users. This type of VPN connection uses the OpenVPN protocol and is typically accessed using a client application installed on the user's device.
- **AWS Transit Gateway VPN:** Enables you to create a secure connection between your VPC and an on-premises network, as well as connections between multiple VPCs and on-

premises networks. This type of VPN connection uses IPsec to encrypt the data transmitted between the networks.

- **AWS VPN CloudHub:** With CloudHub, multiple remote sites can communicate with the VPC and each other. CloudHub design follows the traditional hub-and-spoke model.

Deploying AWS VPN CloudHub, on the AWS side, there is a single VPG; however, there are multiple customer gateways required as there are multiple connection paths from multiple physical sites (see [Figure 4-22](#)). Each customer gateway requires a unique BGP ASN to distinguish its location. The maximum bandwidth of each AWS VPN connection at AWS is 1.25 Gbps.



**Figure 4-22** VPN CloudHub Design Choices

## Understanding Route Propagation

After route table entries have been created to allow VPN connections from the customer gateway, you can enable the automatic provisioning of the available routes through *route propagation*. To enable automatic route propagation, choose the Route Propagation tab from the properties of the route table and then select the VPG to assign to the route table. Route propagation allows a virtual private gateway to automatically propagate routes to the route tables, ensuring efficient communications.

Each AWS VPN connection created in AWS has two tunnels for failover on the Amazon side of the connection. Each tunnel has a unique security association (SA) that identifies each tunnel's inbound and outbound traffic. If static routes are available, when an AWS VPN connection is activated, the static addresses for your customer data center and the CIDR ranges for the connected VPC are automatically added to the route table.

## AWS Direct Connect

AWS Direct Connect is a service provided by AWS that enables you to establish a dedicated network connection from your on-

on-premises data center to AWS. It offers two types of connections:

- **Dedicated connection:** This type of connection provides a dedicated, single-tenant network connection between your on-premises network and your VPC. The dedicated connection uses a physical network connection with a capacity of from 1 to 100 Gbps.
- **Hosted connection:** This type of connection enables you to establish a connection to AWS Direct Connect over the public Internet. The hosted connection uses a virtual interface with a capacity of 50 Mbps, 100 Mbps, or 200 Mbps.

Each AWS Direct Connect dedicated connection ordered is a single dedicated connection from your organization's routers to an AWS Direct Connect router. Virtual interface connections can be created to connect directly to AWS services or VPCs. A virtual public interface enables access to Amazon cloud services; a private virtual interface enables access to a VPC.

AWS Direct Connect dedicated connections support 1000BASE-LX or 10GBASE-LR connections over single-mode fiber using Ethernet transport and 1310 nm connectors.

A hosted connection is provided by an AWS Direct Connect partner from the customer data center to the facility where

AWS Direct Connect connections can be made. The connection speeds available from the selected Amazon partner can range from 1 to 100 Gbps. To sign up for AWS Direct Connect, open the AWS Direct Connect Dashboard and complete the following steps:

**Step 1.** Request a connection, specifying the port speed and the Direct Connect location where the connection will be terminated. If your port speed required is less than 1 Gbps, you must contact a registered AWS Direct Connect vendor that is a member of the Amazon Partner Network (APN) in your geographic location and order a hosted connection at the bandwidth you desire. When this connection is complete, the setup of Direct Connect can continue in the console.

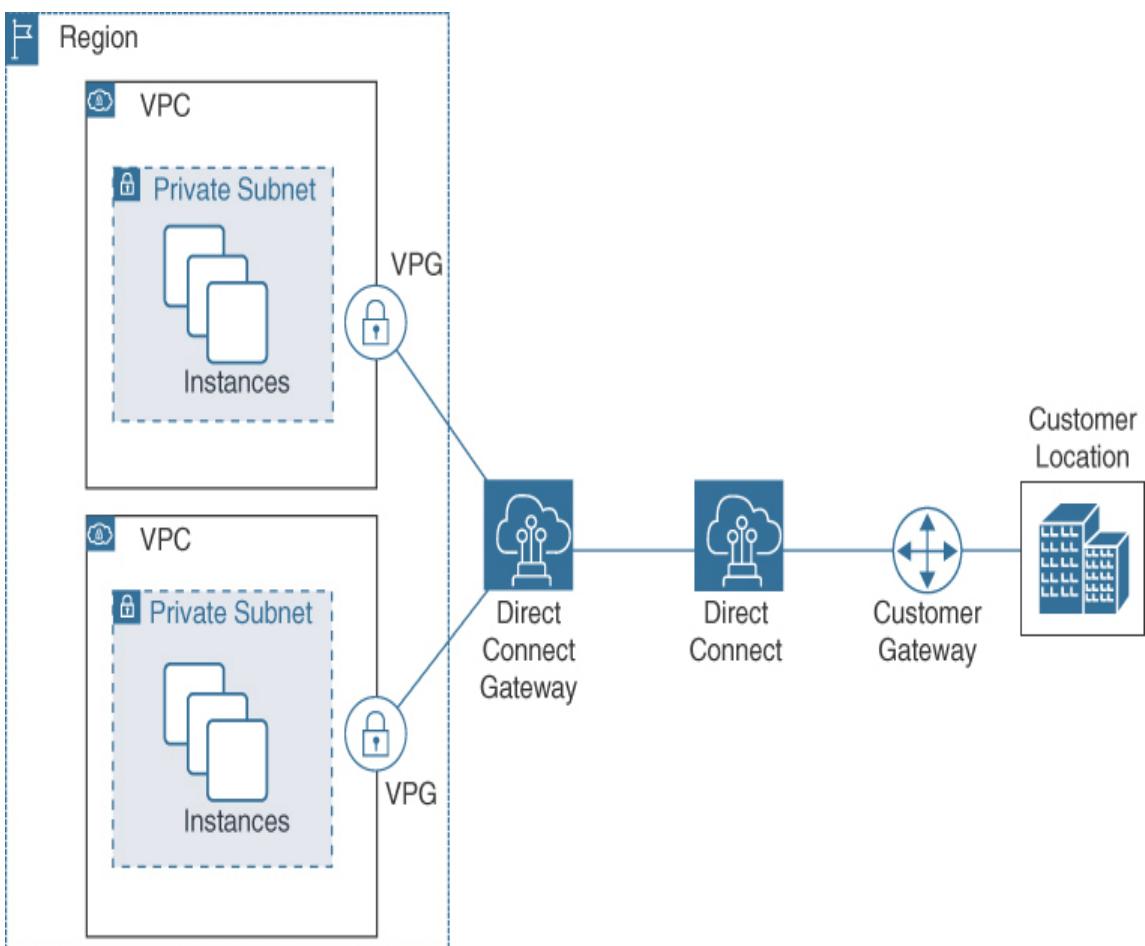
**Step 2.** When AWS has approved your connection, download a Letter of Authorization-Connecting Facility Assignment (LOA-CFA) and present it to your provider as authorization to create a cross-connect network connection to AWS.

**Step 3.** Create virtual interfaces for the required connections to either a VPC or a public AWS service.

**Step 4.** After virtual interfaces have been created, download the router configuration file containing detailed router

configuration information to successfully connect to the virtual interfaces.

There are many considerations for AWS Direct Connect, including your location, the AWS region you are operating in, the level of redundancy required, the number of VPCs, public AWS services, or AWS Direct Connect gateways that you connect (see [Figure 4-23](#)).



**Figure 4-23** Direct Connect Choices

## AWS Direct Connect Gateway

A Direct Connect Gateway is a component of AWS Direct Connect that enables you to connect multiple virtual private clouds to a single AWS Direct Connect connection. A Direct Connect gateway acts as a central hub for the VPCs that are connected to it, enabling the routing of traffic between the VPCs.

## AWS Direct Connect Cheat Sheet



For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of Direct Connect:

- You can configure an AWS Direct Connect connection with one or more virtual interfaces (VIFs).
- Public VIFs allow access to services such as Amazon S3 buckets and Amazon DynamoDB tables.
- Private VIFs allow access only to VPCs.
- An AWS Direct Connect connection allows connections to all availability zones within the region where the connection

has been established.

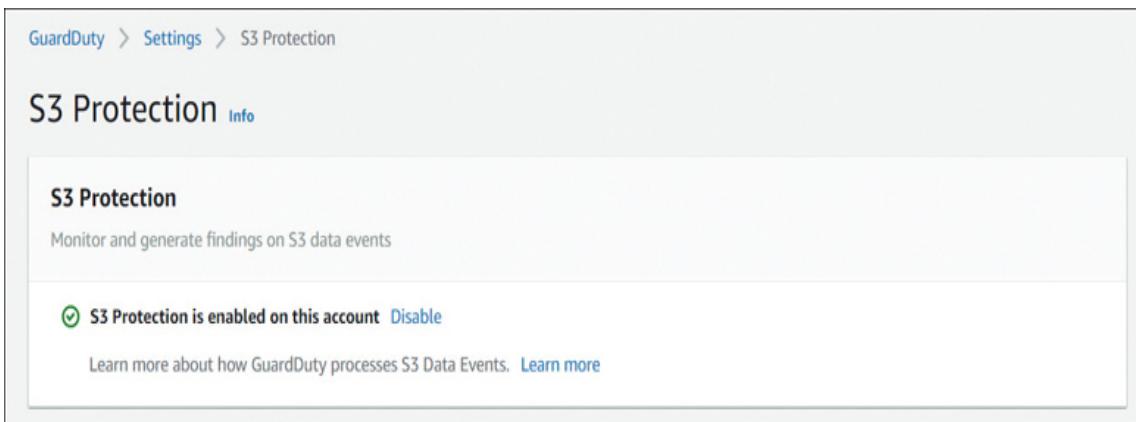
- You are charged for AWS Direct Connect connections based on data transfer and port hours used.
- AWS Direct Connect dedicated connections are available at 1 Gbps up to 100 Gbps speeds.
- You can order speeds of 50 Mbps up to 200 Mbps through a hosted connection through AWS Direct Connect partners.
- An AWS Direct Connect gateway allows you to connect to multiple VPCs.
- An AWS Direct Connect gateway can connect to virtual private gateways and private virtual interfaces owned by the same AWS account.
- An AWS Direct Connect gateway can be associated with AWS Transit Gateway, extending an organization's private network.
- An AWS Direct Connect connection can also be used with an IPsec VPN connection for additional security.

## Amazon GuardDuty



Amazon GuardDuty is a threat detection service that continuously monitors and protects your AWS account, EC2

instances, container applications Amazon Aurora databases, and data stored in S3 buckets (see [Figure 4-24](#)). It uses machine learning and anomaly detection to identify potentially malicious activity in your AWS environment, such as unauthorized access or unusual behavior. GuardDuty provides alerts for any suspicious activity it detects, allowing organizations to take appropriate action to protect AWS resources. AWS GuardDuty also supports AWS Organizations.



**Figure 4-24** GuardDuty Settings

Amazon GuardDuty relies on machine learning anomaly detection, network monitoring and malicious file detection using AWS, and third-party security knowledge to analyze AWS security services, including CloudTrail events, VPC flow logs, Amazon Elastic Kubernetes Service audit logs, and DNS query logs. Amazon GuardDuty performs near-real-time analysis and actions can be automated using AWS Lambda or Amazon

EventBridge. Amazon GuardDuty is helpful when deployments are too extensive for organizations to adequately manage and protect AWS resources.

Once enabled, Amazon GuardDuty starts analyzing account, network, data activity, and AWS services enabled for analysis in near real time. Amazon GuardDuty monitors for many security issues, including the following:

- **Reconnaissance:** Amazon GuardDuty scans for unusual API activity, failed database login attempts using CloudTrail management event logs, and suspicious ingress and egress network traffic using VPC flow logs (see [Figure 4-25](#)).
- **Global events:** Amazon GuardDuty monitors CloudWatch global events for malicious IAM user behaviors, AWS Security Token Service, unauthorized Amazon S3 access, Amazon CloudFront, and Amazon Route 53 for malicious usage across AWS regions.
- **Amazon EC2 instance compromise:** Amazon GuardDuty monitors network protocols, inbound and outbound communication, and compromised EC2 credentials. Amazon GuardDuty Malware Protection, when enabled, scans EBS volumes attached to EC2 instances and container workloads.
- **Amazon EKS Protection:** Amazon GuardDuty monitors Amazon EKS cluster control plane activity by analyzing

Amazon EKS audit logs for issues.

- **Amazon RDS Protection:** Amazon GuardDuty monitors access attempts to existing and new Aurora databases.
- **Amazon S3 Bucket compromise:** Amazon GuardDuty monitors for suspicious data patterns by analyzing AWS CloudTrail management events and Amazon S3 data events, including **Get**, **Put**, **List**, and **Delete** object API operations from a remote host or unauthorized S3 access from known malicious IP addresses.
- **Amazon Route 53 DNS logs:** GuardDuty monitors Amazon Route 53 request and response logs for security issues.

The screenshot shows the AWS GuardDuty Findings interface. At the top, it says "Showing 2 of 2" with three status indicators: blue (2), orange (0), and red (0). Below this is a toolbar with "Findings" (selected), "Info", "Actions", and a search bar. Underneath is a section for "Saved rules" with a note "No saved rules". A "Current" dropdown and a "Add filter criteria" button are also present. The main area displays a table of findings:

Finding type	Resource	Last seen	Count
Policy: IAMUser/RootCredentialUsage	mbw: ASIAUSE3OMLYACGKP7UE	2 hours ago	5507
Stealth: IAMUser/CloudTrailLoggingDisabled	AWSControlTowerAdmin: ASIAJ2ZBDJ2HOENBTLSA	8 days ago	1

**Figure 4-25** GuardDuty Findings

## Amazon GuardDuty Cheat Sheet

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of

## GuardDuty:

- Amazon GuardDuty can also be deployed with AWS Organizations (AWS recommended deployment).
- When Amazon GuardDuty Malware Protection finds issues with EBS volumes, it creates replica snapshots of the affected EBS volumes.
- Amazon GuardDuty can also be integrated with AWS Security Hub and Amazon Detective Services to perform automated actions.

## Amazon Macie

**Key Topic**

Amazon Macie is a security service provided by AWS that uses machine learning to automatically discover, classify, and protect sensitive data stored in S3 buckets. Amazon Macie helps you secure your data and prevent unauthorized access or accidental data leaks.

Amazon Macie uses machine learning and pattern matching to discover and protect sensitive data, such as personally identifiable information (PII) and intellectual property (IP).

Discovered issues are presented as detailed findings for sensitive data, review, and remediation.

Amazon Macie runs data discovery jobs on a schedule or a one-time basis (see [Figure 4-26](#)), which starts the automated discovery, logging, and reporting of any security and privacy issues that are discovered. Each job selects the S3 bucket(s) and bucket criteria (name, account ID, effective permissions, shared access, and tags). Up to 1000 Amazon S3 buckets and AWS accounts can be selected per discovery job. The following sensitive data types are identified using data identifiers:

- Credential data such as private keys or AWS secret access keys
- Credit card and bank account numbers
- Personal information, health insurance details, passports, and medical IDs
- Custom identifiers consisting of regular expressions (regex) per organization, such as employee IDs, or internal data identifiers

The screenshot shows the 'Choose S3 buckets' step in the Macie job creation process. On the left, a sidebar lists steps from 1 to 6. Step 1 is 'Choose S3 buckets', which is highlighted. Step 2 is 'Review S3 buckets', Step 3 is 'Refine the scope', Step 4 is 'Select managed data identifiers', Step 5 is 'Select custom data identifiers', and Step 6 is 'Select allow lists'. The main area is titled 'Choose S3 buckets' with a 'Info' link. It explains that a job can analyze objects in one or more S3 buckets. Two options are shown: 'Select specific buckets' (selected) and 'Specify bucket criteria'. The 'Select specific buckets' section contains a note about manually selecting buckets and analyzing objects in the same buckets each time the job runs. The 'Specify bucket criteria' section contains a note about entering criteria to determine which buckets contain objects for the job to analyze. Below this is a table titled 'Select S3 buckets (1/25+)'. The table header includes columns for Bucket, Account, Classifiable objects, Classifiable files, Monitored by, and Latest job run. It shows two rows: one for '313858614000-awsmacietrail-data...' and another for '3632535253523255', where the second row has a checked checkbox in the Bucket column.

Bucket	Account	Classifiable objects	Classifiable files	Monitored by...	Latest job run
313858614000-awsmacietrail-data...	313858614...	100.3 k	148.7 MB	No	
3632535253523255	313858614...	1	344.5 KB	No	

**Figure 4-26** Amazon Macie Job Configuration

Amazon Macie data findings are published to the Amazon Macie console. Amazon EventBridge events can be configured that call a custom AWS Lambda function to perform automated remediation tasks.

## Amazon Macie Cheat Sheet

For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of Amazon Macie:

- AWS Organizations uses multiple Amazon Macie accounts: an Administrator account that manages the Amazon Macie

accounts for the organization and member accounts.

- Sensitive data can be identified using a custom data identifier or keyword.
- Amazon Macie can publish sensitive data policy findings automatically to Amazon EventBridge as events.
- A policy finding provides a detailed report of a potential policy violation (for example, unexpected access to S3 bucket), including a severity rating, detailed information, and when the issue was found.
- Amazon Macie publishes near-real-time logging data to CloudWatch logs.
- Amazon Macie can analyze encrypted objects with the exception of objects encrypted with customer-provided keys (SSE-C).

## Security Services for Securing Workloads



For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the use cases for the following AWS security tools for monitoring and managing hosted workloads:

- AWS CloudTrail
- AWS Secrets Manager
- Amazon Inspector
- AWS Trusted Advisor
- AWS Config

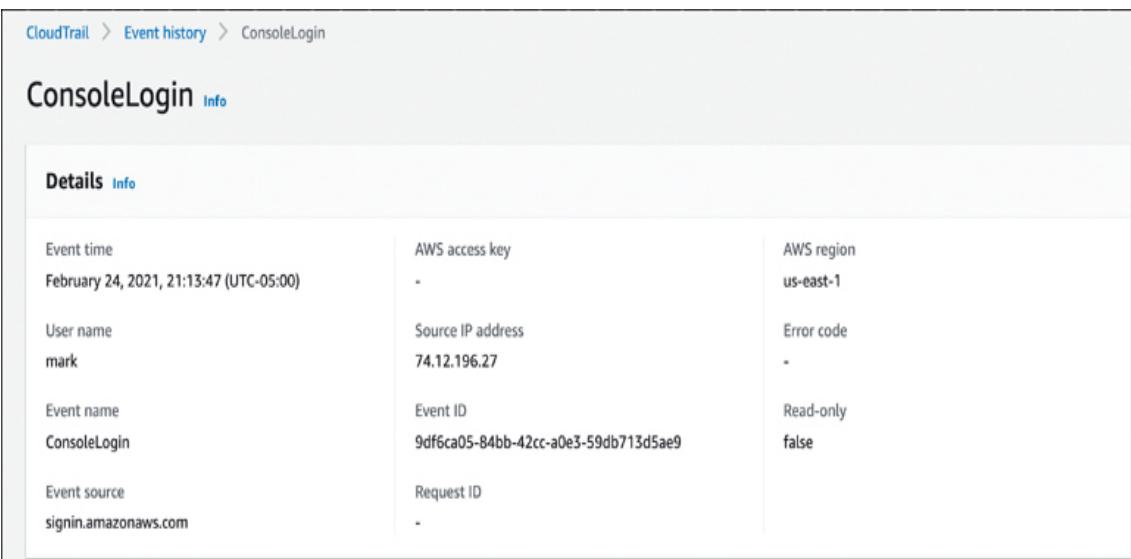
## AWS CloudTrail

AWS CloudTrail records all AWS API calls carried out within each AWS account related to actions across your AWS infrastructure. AWS CloudTrail also logs all account authentications and event history for your AWS account, including actions taken through the AWS Management Console, AWS SDKs, the AWS CLI command prompt, and other AWS service activity. AWS CloudTrail is enabled by default for all AWS accounts, and events in the default CloudTrail trail are available for the last 90 days free of charge. Organizations can also create custom trails storing all activity indefinitely in an Amazon S3 bucket or Amazon CloudWatch log group. Amazon CloudWatch events are logged in AWS CloudTrail within 15 minutes of each API request. The following are common tasks that CloudTrail is useful for:

- Review event history and insights for resource management, compliance, and operational and risk auditing.

- Review event history in AWS CloudTrail for information on successful and unsuccessful authentication requests.
- Review API calls carried out in an AWS account.

[\*\*Figure 4-27\*\*](#) shows the details of an AWS management console logon listing Amazon S3 Buckets by IAM user Mark, including the AWS account ID, username, time, source, and region.



The screenshot shows a CloudTrail event history page. The top navigation bar includes 'CloudTrail' > 'Event history' > 'ConsoleLogin'. Below this, the event details are displayed under the heading 'ConsoleLogin [Info](#)'. A 'Details' tab is selected, showing the following data in a table:

Details <a href="#">Info</a>		
Event time February 24, 2021, 21:13:47 (UTC-05:00)	AWS access key -	AWS region us-east-1
User name mark	Source IP address 74.12.196.27	Error code -
Event name ConsoleLogin	Event ID 9df6ca05-84bb-42cc-a0e3-59db713d5ae9	Read-only false
Event source signin.amazonaws.com	Request ID -	

**Figure 4-27** Detailed CloudTrail Event

AWS CloudTrail is a regional service with a global reporting reach because the default trail automatically creates separate trails in each active AWS region. AWS CloudTrail events for each AWS region can be viewed using the AWS CloudTrail console and manually switching to the desired AWS region. IAM policies can be created using the AWS Identity and Access

Management service to control which IAM users can create, configure, or delete AWS CloudTrail trails and events.

## Creating an AWS CloudWatch Trail

To store AWS CloudTrail events longer than the default 90-day time frame, create a custom trail that stores the AWS CloudTrail event information in an Amazon S3 bucket or Amazon CloudWatch log group. Management read/write events, or just read-only or write-only events, can be added to your custom trail, as shown in [Figure 4-28](#). Optionally, you can also create an AWS Simple Notification topic to receive notifications when specific events have been delivered to the Amazon CloudWatch log group.

**General details**

A trail created in the console is a multi-region trail. [Learn more](#)

**Trail name**  
Enter a display name for your trail.

Audit\_2021

3-128 characters. Only letters, numbers, periods, underscores, and dashes are allowed.

**Enable for all accounts in my organization**

To review accounts in your organization, open AWS Organizations. [See all accounts](#)

**Storage location** [Info](#)

**Create new S3 bucket**  
Create a bucket to store logs for the trail.

**Use existing S3 bucket**  
Choose an existing bucket to store logs for this trail.

**Trail log bucket and folder**  
Enter a new S3 bucket name and folder (prefix) to store your logs. Bucket names must be globally unique.

aws-cloudtrail-logs-313858614000-aee218e7

Logs will be stored in aws-cloudtrail-logs-313858614000-aee218e7/AWSLogs/o-bqSyhpe6ls/313858614000

**Log file SSE-KMS encryption** [Info](#)

**Enabled**

**Figure 4-28** Creating a CloudTrail Trail

After data has been logged and stored in a custom trail, analysis and possible remediation can be performed using these methods:

- **Amazon S3 bucket:** API activity for the S3 bucket can trigger a notification to an AWS SNS topic or trigger an AWS Lambda custom function.
- **AWS Lambda function:** Custom AWS Lambda functions can respond to selected AWS CloudTrail data events.

- **AWS CloudTrail Insights:** CloudTrail Insights can be used to detect unusual activity for individual CloudTrail write management events within an AWS account.
- **AWS CloudTrail events:** A CloudTrail event can display a specific pattern, such as authentication as the root user, as shown in [Figure 4-29](#).

ConsoleLogin <a href="#">Info</a>	
Details <a href="#">Info</a>	
Event time	AWS access key
January 31, 2021, 12:21:25 (UTC-05:00)	-
User name	Source IP address
root	50.101.23.166
Event name	Event ID
ConsoleLogin	2eb71d03-8466-4aa3-be23-c4e97653ec45
Event source	Request ID
signin.amazonaws.com	-

**Figure 4-29** CloudTrail Authentication Event

## AWS CloudTrail Cheat Sheet



For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of

## AWS CloudTrail:

- AWS CloudTrail records all activity on an AWS account, including API calls and authentications.
- Custom AWS CloudWatch trails can deliver events to an S3 bucket or a CloudWatch log group.
- AWS CloudTrail events can be used for auditing AWS account activity.
- AWS CloudTrail reports activity for each AWS account.
- AWS CloudTrail can be integrated with an AWS Organization.
- AWS CloudTrail tracks both data and management events.
- AWS CloudTrail records can be encrypted using S3 server-side encryption.

## AWS Secrets Manager



AWS Secrets Manager is a service that enables you to store, rotate, and manage organizational secrets used to access your applications, services, and IT resources. With Secrets Manager, you can securely store and manage secrets, such as database credentials and API keys, helping reduce the risk of secrets being compromised and meet compliance requirements. AWS

Secrets Manager enables you to secure and manage secrets for SaaS applications, SSH keys, RDS databases, third-party services, and on-premises resources (see [Figure 4-30](#)). You can also store credentials for MySQL, PostgreSQL, and Amazon Aurora, and Oracle databases hosted on EC2 instances and OAuth refresh tokens used when accessing third-party services and on-premises resources.

Select secret type [Info](#)

Credentials for RDS database  Credentials for DocumentDB database  Credentials for Redshift cluster

Credentials for other database  Other type of secrets (e.g. API key)

Specify the user name and password to be stored in this secret [Info](#)

User name  
mark

Password  
.....  
 Show password

**Figure 4-30** Storing RDS Credentials as a Secret

When database secrets are stored in AWS Secrets Manager, the rotation of database credentials can be automatically configured. Secrets are encrypted at rest using encryption keys stored in AWS Key Management Service. You can either specify

customer master keys (CMKs) to encrypt secrets or use the default AWS KMS encryption keys provided for your AWS account.

---

#### Note

Use of the term *master* is ONLY in association with the official terminology used in industry specifications and/or standards, and in no way diminishes Pearson's commitment to promoting diversity, equity, and inclusion, and challenging, countering, and/or combating bias and stereotyping in the global population of the learners we serve.

---

Using the AWS Secrets Manager APIs, developers can replace any hard-coded secrets used in their applications with secrets retrieved from Secrets Manager. Access to secrets is controlled by the IAM policy, which defines the access permissions of users and applications when retrieving secrets.

Applications that are running on EC2 instances hosted within a VPC can use a private interface endpoint to connect directly to AWS Secrets Manager across the AWS private network.

## **Amazon Inspector**

Amazon Inspector allows you to test the security levels of instances you have deployed. After you define an assessment target for Amazon Inspector, which is a group of tagged EC2 instances, Amazon Inspector evaluates the state of each instance by using several rule packages.

Amazon Inspector uses two types of rules: network accessibility tests that don't require the Inspector agent to be installed, and host assessment rules that require the Inspector agent to be installed (see [Figure 4-31](#)). Amazon Inspector performs security checks and assessments against the operating systems and applications hosted on Linux and Windows EC2 instances by using an optional Inspector agent installed on the operating system associated with the EC2 instance.

Assessment Template - Feb 2021

Name\* Feb 2021

Target name\* Corporate Web Servers

Rules packages\* CIS Operating System Security Configuration Benchmarks-1.0 x

Select an Inspector rules package

Duration\* 1 Hour (Recommended)

SNS topics Select a new SNS topic to notify of events

Topic	Events
313858614000:ec2_instance_changes	Run started <span style="color: red;">x</span> Run finished <span style="color: red;">x</span> Run state changed <span style="color: red;">x</span> Finding reported <span style="color: red;">x</span>

**Figure 4-31** Amazon Inspector Options

Assessment templates check for any security issues on targeted EC2 instances. The choices for rule packages comply with industry standards. They include Common Vulnerabilities and Exposure (CVE) checks, Center for Internet Security (CIS) checks, operating system configuration benchmarks, and other security best practices. Current supported levels of CVE checks can be found at <https://nvd.nist.gov/general>. The Amazon Inspector Network Reachability rules package allows you to identify ports and services on your EC2 instances that are reachable from outside the VPC. Amazon Inspector gathers the current network configuration, including security groups,

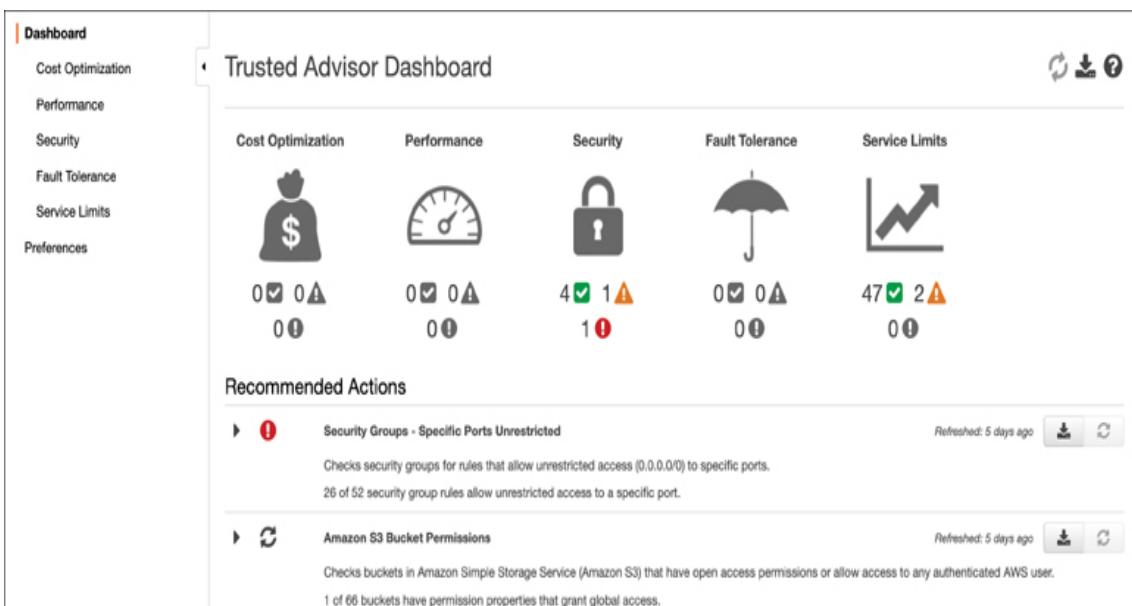
network access control lists, and route tables, and analyzes the accessibility of the instance.

Amazon Inspector rules are assigned severity levels of medium and high based on the defined assessment target's confidentiality, integrity, and availability. Amazon Inspector also integrates with Amazon Simple Notification Service (SNS), which sends notifications when failures occur. An AWS SNS notification can, in turn, call an AWS Lambda function, which can carry out any required task; AWS Lambda can call any AWS API. Amazon Inspector can alert you when security problems are discovered on web and application servers, including insecure network configurations, missing patches, and potential vulnerabilities in the application's runtime behavior.

## AWS Trusted Advisor

AWS Trusted Advisor is a built-in management service that executes several essential checks against your AWS account resources (see [Figure 4-32](#)). Every AWS account has access to several core AWS Trusted Advisor checks, and access to the AWS Personal Health Dashboard, which alerts you when specific resources you are using at AWS are having issues. The following are core AWS Trusted Advisor checks:

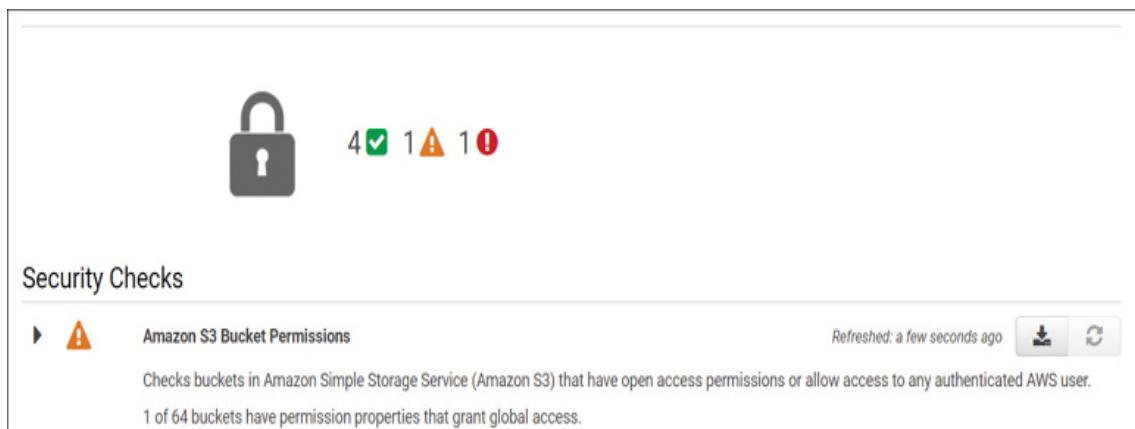
- Security checks include permission checks for EBS and RDS snapshots that are marked public, S3 buckets that have open access, checking for the creation of at least one IAM user, and root accounts that don't have MFA enabled.
- AWS Trusted Advisor checks various AWS services and alerts when usage is greater than 80% of the current service quota limits in force, including IAM users, Amazon S3 buckets created, VPCs created, and Auto Scaling groups.



**Figure 4-32** Trusted Advisor Security Checks

AWS Trusted Advisor can also provide additional checks if your organization has purchased Business or Enterprise support. Full AWS Trusted Advisor checks provide recommendations for improving performance, security, fault tolerance, and cost-

effectiveness. AWS Trusted Advisor is useful to run against AWS account resources to review current security issues and any flagged service quotas (see [Figure 4-33](#)).



**Figure 4-33** Trusted Advisor Security Checks Results

Once Business and Enterprise support has been purchased, AWS Trusted Advisor can alert you about issues to check, including the following:

- **Reserved Amazon EC2 instances:** AWS Trusted Advisor can calculate the optimized number of partial upfront reserved instances required based on an analysis of your usage history for the past month.
- **AWS ELB load balancers:** AWS Trusted Advisor checks current AWS ELB load balancing usage.
- **EBS volume check:** AWS Trusted Advisor warns if AWS EBS volumes in your AWS account are unattached or have low

access rates.

- **Elastic IP addresses:** AWS Trusted Advisor warns if any Elastic IP addresses assigned to your account have not been associated. (Charges apply if Elastic IP addresses in your account are not used.)
- **Amazon RDS instances:** AWS Trusted Advisor checks for idle AWS RDS database instances.
- **Amazon Route 53 records:** AWS Trusted Advisor checks whether the creation of latency record sets has been properly designed to replicate end-user requests to the best AWS region.
- **Reserved reservation expiration check:** AWS Trusted Advisor warns you if your current reserved reservation is scheduled to expire within the next month. (Reserved reservations do not automatically renew.)

## AWS Config

AWS Config enables customers to monitor, audit, and evaluate the deployed configurations of deployed IaaS resources, including EC2 instances, VPCs and components, IAM permissions, and S3 buckets deployed in a single AWS account or AWS accounts managed by an AWS organization. AWS Config provides detailed records of resource inventory, configuration

history, and changes. Configuration data collected by AWS Config is stored in Amazon S3 buckets and Amazon DynamoDB.

The following are features of AWS Config:

- **Resource inventory:** Up-to-date inventory of selected AWS resources is recorded on an automated schedule.
- **Configuration History:** Configuration changes to AWS resources are tracked and stored, providing a historical view of changes over time.
- **Configuration Compliance:** Resources can be evaluated against predefined or custom rules, assessing the compliance of deployed AWS infrastructure components.
- **Management of Resources:** Centrally storing AWS resources helps an organization manage compliance and security standards.
- **Rules:** Managed rules created by AWS (see [Figure 4-34](#)) and custom rules can be used to evaluate resource configurations against predefined or custom criteria. Organizations can create their own custom AWS Config rules based on specific governance requirements such as security policies, compliance standards, or adhering to best practices. Custom rules are created using the AWS Lambda functions. Resource-specific rules could be created to evaluate the capacity of EC2

instances, the configuration of S3 buckets, or the configuration of a VPC.

- **Event Management:** SNS events can be generated when resource configurations change.

Rules				
Rules represent your desired configuration settings. AWS Config evaluates whether your resource configurations comply with relevant rules and summarizes the compliance results.				
Rules		Actions	Add rule	
Any status			< 1 2 > ⌂	
Name	Remediation action	Type	Enabled evaluation mode	Detective compliance
s3-bucket-versioning-enabled	Not set	AWS managed	DETECTIVE	⚠ 18 Noncompliant resource(s)
encrypted-volumes	Not set	AWS managed	DETECTIVE	-
access-keys-rotated	Not set	AWS managed	DETECTIVE	⚠ 15 Noncompliant resource(s)
root-account-mfa-enabled	Not set	AWS managed	DETECTIVE	⌚ Compliant
cloud-trail-encryption-enabled	Not set	AWS managed	DETECTIVE	⚠ 2 Noncompliant resource(s)

**Figure 4-34** AWS Config Managed Rules

## Exam Preparation Tasks

As mentioned in the section “[How to Use This Book](#)” in the Introduction, you have a couple of choices for exam preparation: the exercises here, [Chapter 16](#), “[Final Preparation](#),” and the exam simulation questions in the Pearson Test Prep Software Online.

## Review All Key Topics

Review the most important topics in the chapter, noted with the Key Topic icon in the margin of the page. [Table 4-9](#) lists these key topics and the page number on which each is found.

**Table 4-9** [Chapter 4](#) Key Topics

Key Topic Element	Description	Page Number
<a href="#">Figure 4-1</a>	Connections and Security Services	150
Section	AWS Shield (Standard and Advanced)	151
Section	AWS Web Application Firewall (WAF)	152
Section	The Main Route Table	155
Section	Custom Route Tables	155

Key Topic Element	Description	Page Number
Section	Route Table Cheat Sheet	158
Section	Security Groups	158
Section	Security Groups Cheat Sheet	161
Section	Understanding Ephemeral Ports	165
<u>Figure 4-11</u>	Security Group Design	168
Section	Network ACLs	168
Section	Network ACL Cheat Sheet	169
Section	VPC Flow Logs	172
Section	NAT Services	174

Key Topic Element	Description	Page Number
Section	AWS NAT Gateway Service Cheat Sheet	176
Section	Amazon Cognito	176
Section	External Connections	180
Section	AWS Direct Connect Cheat Sheet	187
Section	Amazon GuardDuty	187
Section	Amazon Macie	189
Section	Security Services for Securing Workloads	191
Section	AWS CloudTrail Cheat Sheet	194

Key Topic Element	Description	Page Number
Section	AWS Secrets Manager	194

## Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

[AWS Direct Connect](#)

[security group \(SG\)](#)

[network access control list \(NACL\)](#)

[NAT gateway service](#)

## Q&A

The answers to these questions appear in [Appendix A](#). Use the Pearson Test Prep Software Online for more practice with exam format questions.

- 1.** What AWS networking services can replace existing hardware devices?
- 2.** What can network ACLs do that a security group cannot do?
- 3.** What is the benefit of using CloudTrail trails for all AWS regions?
- 4.** What is the benefit of using AWS Secrets Manager?
- 5.** What type of artificial intelligence is used to operate GuardDuty?
- 6.** How can Direct Connect help with high-speed connections to multiple VPCs?
- 7.** For what assessments are you not required to have the Amazon Inspector agent installed?
- 8.** How do you enable all checks for Trusted Advisor?

# Chapter 5

## Determining Appropriate Data Security Controls

This chapter covers the following topics:

- [Data Access and Governance](#)
- [Amazon EBS Encryption](#)
- [Amazon S3 Bucket Security](#)
- [AWS Key Management Service](#)
- [AWS Certificate Manager](#)

This chapter covers content that's important to the following exam domain and task statement:

Domain 1: Design Secure Architectures

Task Statement 3: Determine appropriate data security controls

Organizations have workloads and associated cloud services fail while operating at AWS. Amazon Elastic Compute Cloud (EC2) instances fail, Amazon Elastic Block Store (EBS) volumes crash, and cloud services can stop working. However, you shouldn't have go to your boss and announce, "We've lost some

data.” Fortunately, all data can be securely and redundantly stored at AWS.

All data stored at AWS using any storage service can be encrypted; organizations make the decision about whether encryption is required. However, Amazon S3 objects and S3 Glacier archive storage *is* automatically encrypted at rest. All other storage services at AWS store data records in an unencrypted state to start. For example, Amazon S3 buckets are encrypted using server-side encryption using Amazon S3, the AWS Key Management Service (KMS) with customer master keys (CMK) and data keys, or encryption keys supplied by each organization. Amazon EBS volumes—both boot and data volumes—can be encrypted at rest and in transit using CMKs provided by AWS KMS. Shared storage services such as Amazon EFS and Amazon FSx for Windows File Server can also be encrypted at rest, as can Amazon DynamoDB tables, Amazon Relational Database Service (RDS) deployments, and Amazon Simple Queue Service (SQS) queues.

---

#### Note

Use of master/slave terms in the following chapter is ONLY in association with the official terminology used in industry specifications and/or standards,

and in no way diminishes Pearson’s commitment to promoting diversity, equity, and inclusion, and challenging, countering, and/or combating bias and stereotyping in the global population of the learners we serve.

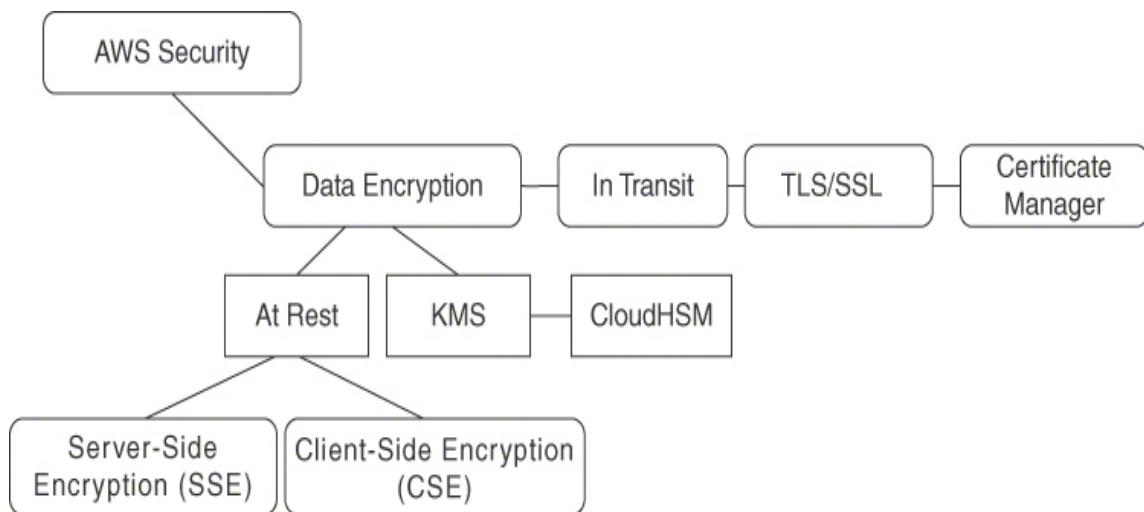
---

AWS does not have single-tenant persistent data storage for individual organizations; all storage services offered at AWS are multi-tenant by design. AWS has the responsibility to ensure that each organization’s stored data records are isolated to the AWS account in which they are first created. Organizations can secure data at rest by choosing to encrypt all data records; protecting data in transit can be achieved using Transport Layer Security (TLS).

Each organization is in control of the storage and retrieval of its data records that are stored at AWS. It’s the organization’s responsibility to define the security and accessibility of all data records stored at AWS. All data storage at AWS starts as private storage only accessible across the AWS private network. Organizations can choose to make select Amazon S3 buckets public, but all other storage services offered by AWS remain private and are not publicly accessible across the Internet. AWS VPN and AWS Direct Connect connections from on-premises

locations can directly access AWS storage services; however, EBS volumes can only be accessed through the attached EC2 instance. [Figure 5-1](#) illustrates the options for data encryption at AWS that are discussed in this chapter.

### Key Topic



**Figure 5-1** Encryption Choices at AWS

### “Do I Know This Already?”

The “Do I Know This Already?” quiz allows you to assess whether you should read this entire chapter thoroughly or jump to the “Exam Preparation Tasks” section. If you are in doubt about your answers to these questions or your own assessment of your knowledge of the topics, read the entire

chapter. [Table 5-1](#) lists the major headings in this chapter and their corresponding “Do I Know This Already?” quiz questions. You can find the answers in [Appendix A, “Answers to the ‘Do I Know This Already?’ Quizzes and Q&A Sections.”](#)

**Table 5-1** “Do I Know This Already?” Section-to-Question Mapping

Foundation Topics Section	Questions
Data Access and Governance	1, 2
Amazon EBS Encryption	3, 4
Amazon S3 Bucket Security	5, 6
AWS Key Management Service	7, 8
AWS Certificate Manager	9, 10

---

### Caution

The goal of self-assessment is to gauge your mastery of the topics in this chapter. If you do not

know the answer to a question or are only partially sure of the answer, you should mark that question as wrong for purposes of the self-assessment.

Giving yourself credit for an answer you correctly guess skews your self-assessment results and might provide you with a false sense of security.

---

**1.** What AWS service assists in protecting access to AWS?

1. AWS Shield
2. Amazon Macie
3. Amazon EBS volumes
4. Amazon DynamoDB databases

**2.** What is the purpose of using detective controls?

1. To enable and enforce multifactor access
2. To detect and alert when security controls change
3. To manage AWS Organizations backups
4. To analyze compliance levels

**3.** Which of the following determines whether an attached Amazon EBS volume can be encrypted?

1. The type of Amazon EC2 instance

2. The size of the Amazon EBS volume
3. The type of the Amazon EBS volume
4. The IOPS assigned to the Amazon EBS volume

**4.** Where are data keys stored when they are delivered to an Amazon EC2 instance for safekeeping?

1. The associated Amazon EBS volume
2. Unsecured RAM
3. Secured RAM
4. AWS Key Management Service

**5.** What security policy allows multiple AWS accounts to access the same Amazon S3 bucket?

1. Amazon IAM policy
2. AWS IAM server control policy
3. Amazon S3 Bucket policy
4. Amazon IAM policy

**6.** What type of encryption can be carried out before uploading objects to Amazon S3 to ensure absolute encryption outside AWS control?

1. RSA encryption
2. AES 128-bit encryption

- 3. Client-side encryption
- 4. Server-side encryption

**7.** What is the advantage of importing your organization's symmetric keys into AWS KMS?

- 1. High level of compliance
- 2. Faster encryption and decryption
- 3. Absolute control of encryption keys
- 4. None

**8.** What additional AWS service can work with AWS KMS as a custom key store?

- 1. Encrypted EBS volume
- 2. Encrypted Amazon S3 bucket
- 3. AWS CloudHSM
- 4. Encrypted AWS SQS queue

**9.** How does AWS charge for provisioning SSL/TLS certificates for AWS services using AWS Certificate Manager?

- 1. It charges per certificate per year.
- 2. It charges for private TLS certificates only.
- 3. It does not charge for AWS services.
- 4. It charges per certificate check.

**10.** Where are the security certificates for the AWS Application Load Balancer stored?

1. Amazon S3 bucket
2. Amazon EBS volume
3. AWS Certificate Manager
4. AWS KMS service

## Foundation Topics

### Data Access and Governance

Many on-premises and AWS-hosted workloads store their associated data records in the AWS cloud. Personal data stored in the public cloud is sometimes defined as personally identifiable information (PII). Sensitive data types, such as PII, must be protected to comply with privacy regulations such as the General Data Protection Regulation (GDPR), laws such as the Health Insurance Portability and Accountability Act (HIPAA), and industry standards such as the Payment Card Industry Data Security Standard (PCI DSS). More than 13 billion data records have been stolen since 2013, according to the *2022 Thales Data Threat Report* (<https://cpl.thalesgroup.com/data-threat-report>). AWS Artifact, located in the AWS Management console, provides on-demand access to all current AWS compliance and

security reports, including Service Organization Control (SOC) and Payment Card Industry (PCI) reports and certifications from accreditation bodies validating the implementation and operating effectiveness of AWS security controls (see [Figure 5-2](#)).

The screenshot shows the AWS Artifact interface with the path 'AWS Artifact > Reports'. Below the navigation bar, there is a search bar containing the query 'pci' with a result count of '4 matches'. A table lists four reports, with the first one expanded to show its details. The expanded report row includes columns for Title, Reporting period, Category, and Description.

Title	Reporting period	Category	Description
PCI 3DS Attestation of Compliance (AOC) and Responsibility Summary - Current	July 15, 2022 to July 14, 2023	Certifications and Attestations	This is the most recent AWS PCI 3DS assessment package dated June 30, 2022. An external Qualified Security Assessor Company (QSA), Coalfire Systems Inc. has validated that AWS has successfully completed PCI 3DS Core Security Standard v1.0 assessment and were found to be compliant.

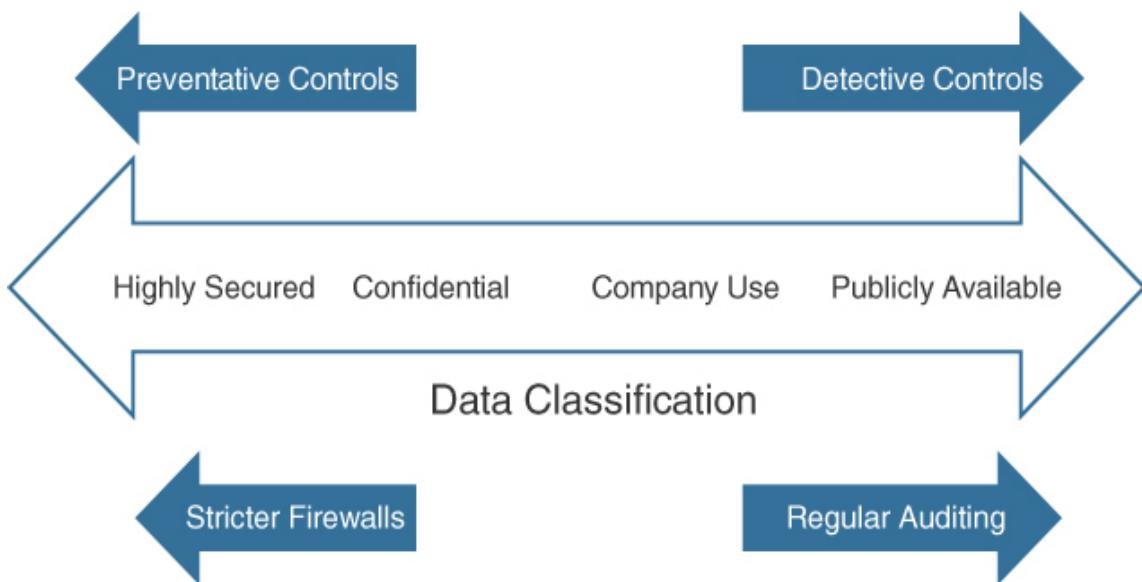
**Figure 5-2** AWS Artifact PCI Report

## Data Retention and Classification



When classifying data, it's important for each organization to implement data retention policies for each class of stored data. Organizations should design security policies using security

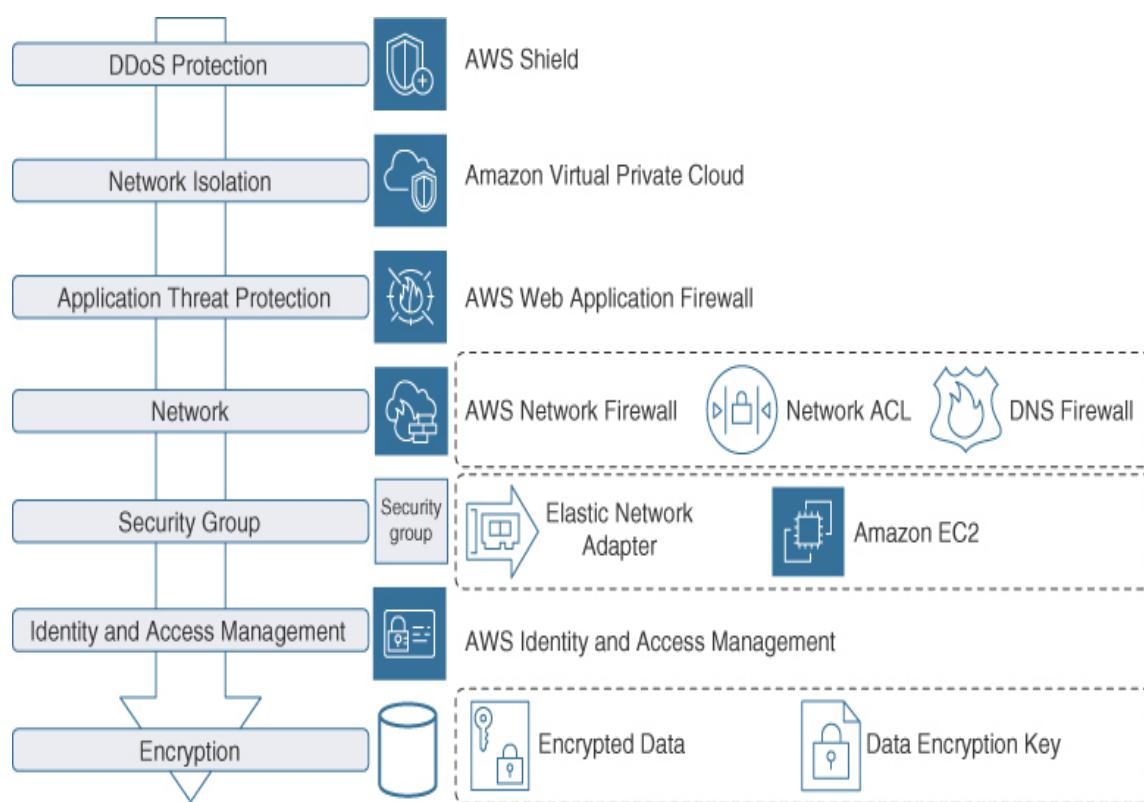
zones for all data records, and data classification requirements based on how data is stored and who has access to it (see [Figure 5-3](#)). Defined security zones for data records range from highly protected to publicly accessible.



**Figure 5-3** Classification of Data Records

Security zones are typically used to segregate different types of organizational data assets based on their sensitivity or importance, with the most sensitive or valuable data being placed in the highest security zone. This segregation enables organizations to implement different levels of security controls and access restrictions based on the sensitivity of the data, ensuring that only authorized users with the appropriate level of clearance can access and view sensitive data records.

Additionally, the creation of relevant security zones can help organizations prevent the spread of security breaches by limiting the potential impact to a specific area of the organization. Organizations also should create a network perimeter with defined network flow and access policies for data records defining where and how data can be accessed. Defense-in-depth security at AWS is applied using infrastructure security controls, AWS IAM security policies, and AWS detective controls (see [Figure 5-4](#)).



**Figure 5-4** Preventative Controls

## Infrastructure Security



Infrastructure security requires deploying the following protections:

- **DDoS Protection:** Amazon deploys AWS WAF and Shield to protect the AWS cloud from DDoS attacks.
- **Network isolation:** EC2 instances must be hosted in a virtual private cloud (VPC). Many AWS services can be accessed from a VPC with private VPC endpoints (Interface and Gateway endpoints), ensuring workload traffic remains on the private AWS network.
- **Application-layer threat protection:** The AWS Web Application Firewall (WAF) allows organizations to create rules and filters to accept or reject incoming requests to Amazon CloudFront distributions, Amazon API Gateway deployments, and Application Load Balancers, and HTTP/HTTPS traffic to web servers.
- **Security groups:** Security groups must be designed to allow ingress traffic from associated security groups.
- **Network ACL:** Design network ACLs to implement zone-based models for your workload (web/app servers/database),

allowing only legitimate traffic to reach each subnet.

## IAM Controls

AWS Identity and Access Management (IAM) policies are useful for controlling access to the data layer (database, queue, AWS EBS volumes, shared data [AWS EFS and AWS FSx for Windows File Server], and Amazon S3 storage) and managing IAM user and federated user activity and infrastructure security.

Separate administrative tasks should be created for Amazon RDS with IAM policies (see [Example 5-1](#)) that control access to database data records. For authentication and authorization to any workload or organizational data records, enable multifactor authentication (MFA) for all administrators and end users.

### Example 5-1 Administrative Access to Amazon RDS

[Click here to view code image](#)

```
"Version": "2012-10-17",
"Statement": [
{
    "Sid": " Controlled Admin Tasks",
    "Effect": "Allow",
    "Action": [
```

```
        "rds:CreateDBSnapshot",
        "rds:StopDBInstance",
        "rds:StartDBInstance"
    ],
    "Resource": [
        "arn:aws:rds:[AWS_region]:[_AWS_account_id]:snapshot:*",
        "arn:aws:rds:[AWS_region]:[_AWS_account_id]:db:demoDB"
    ]
},
{
    "Sid": "DescribeInstances",
    "Effect": "Allow",
    "Action": "rds:DescribeDBInstances",
    "Resource": "*"
}
]
```

## Detective Controls



Detective controls are a type of security control designed to detect and alert when potential security incidents or breaches occur. Detective controls typically are used with preventive and corrective controls forming a comprehensive security strategy. Examples of detective controls at AWS include intrusion detection systems, and auditing or logging systems that monitor user activity and alert on suspicious behavior. The goal of detective controls is to identify potential security threats or vulnerabilities before they can cause harm, allowing organizations to take appropriate action to prevent or mitigate the impact of a security incident.

Detective controls are an important part of a defense-in-depth security strategy as they provide an additional layer of protection by detecting and responding to potential security threats. Detective controls at AWS include the following security services:

- **VPC Flow Logs:** A feature of Amazon VPC that monitors network traffic at the elastic network interface, subnet, or entire VPC. Captured network traffic can be used for troubleshooting connectivity issues and to check current network access rules.
- **AWS CloudTrail:** Continuously monitor and record API usage and user activity across AWS infrastructure.

- **AWS CloudWatch:** Monitors AWS cloud services such as Amazon RDS databases, EC2 instances, and DynamoDB tables and hosted applications by collecting and tracking metric data, application and operating system log files, and using automated responses to defined alarms.
- **Amazon GuardDuty:** Provides continuous threat detection and analysis of VPC Flow Logs, Amazon Route 53 DNS query logs, and AWS CloudTrail S3 data event logs, and protecting AWS accounts and data stored in Amazon S3 from malicious activity. AWS GuardDuty malware protection can help detect malicious files stored on EBS volumes, protecting attached EC2 instances and Amazon Elastic Kubernetes Service (EKS) clusters.
- **AWS Config:** Detects configuration changes in RDS AWS infrastructure including Amazon RDS, EC2 instances, VPC and database architecture, including security groups, database instances, snapshots, and subnet groups.
- **Amazon Macie:** Uses machine learning and pattern matching to protect Amazon S3 objects and sensitive data types.
- **Access Analyzer for S3:** Monitors Amazon S3 buckets and details public or cross-account access.
- **Amazon Detective:** Graphically analyzes AWS CloudTrail management events, VPC Flow Logs, AWS GuardDuty

findings, and Amazon EKS audit logs to help identify the cause of potential security issues.

## Amazon EBS Encryption

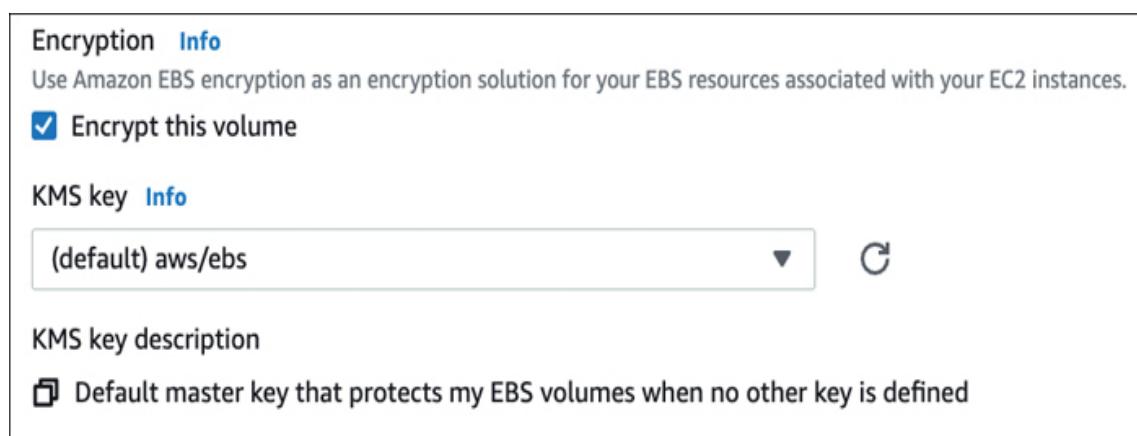


**Amazon Elastic Block Storage (EBS)** volumes provide persistent block-level storage volumes for EC2 instances. They can be used to store a wide variety of data, including operating system files, application data, and database records. EBS volumes are automatically replicated within their availability zone to protect against data loss due to failure, and support a range of performance levels and storage options to meet the needs of different workloads.

Amazon Elastic Block Store (EBS) provides the option to encrypt EBS volumes to protect the data records. Encrypting EBS volumes ensures that the data cannot be read or accessed by unauthorized parties, even if the underlying storage volume is compromised. Encryption is performed using a customer master key and data key managed by the AWS Key Management Service (KMS), which provides a secure and auditable encryption service for managing data encryption at AWS using

encryption keys. EBS volumes can be encrypted when first created, or volumes can be encrypted after they have been created. EBS also provides the option to encrypt snapshots of EBS volumes, enabling you to create encrypted backups of your EBS volumes.

Both EBS boot and data volumes can be encrypted. Most EC2 instances support EBS volumes' encryption, including the C4, I2, I3, M3, M4, R3, and R4 families. AWS has made the encryption process incredibly easy to deploy; when creating an EBS volume, merely checking off the option to enable encryption starts the encryption process (see [Figure 5-5](#)), which is managed by AWS Key Management Service (KMS). More details on AWS KMS are provided throughout this chapter.



**Figure 5-5** Enabling EBS Encryption

---

Note

Data encrypted using the EBS encryption process is encrypted before it crosses the AWS private network. Data also remains encrypted in-flight and at rest and remains encrypted when a snapshot is created of an encrypted volume.

---

The CMK protects all the other keys issued for data encryption and decryption of your EBS volumes within your AWS account. All AWS KMS-issued CMKs are protected using envelope encryption, which means AWS is responsible for creating and wrapping the “envelope” that contains the CMKs of the respective AWS account. Envelope encryption encrypts the plaintext data with a data key, and then encrypts the data key using a key that is managed by the AWS Key Management Service (KMS). KMS keys are created inside AWS KMS and never leave AWS KMS unencrypted. AWS cryptographic tools and services support the Advanced Encryption Standard (AES) with 128-, 192-, or 256-bit keys. AES is combined with Galois/Counter Mode (GCM), which provides high-performance ***symmetric key*** operation using a block size of 128 bits and is used by AWS KMS. AES and GCM are documented as AES-GCM.

After enabling your customer key using KMS for your AWS account, for additional security, it's a good idea to add another

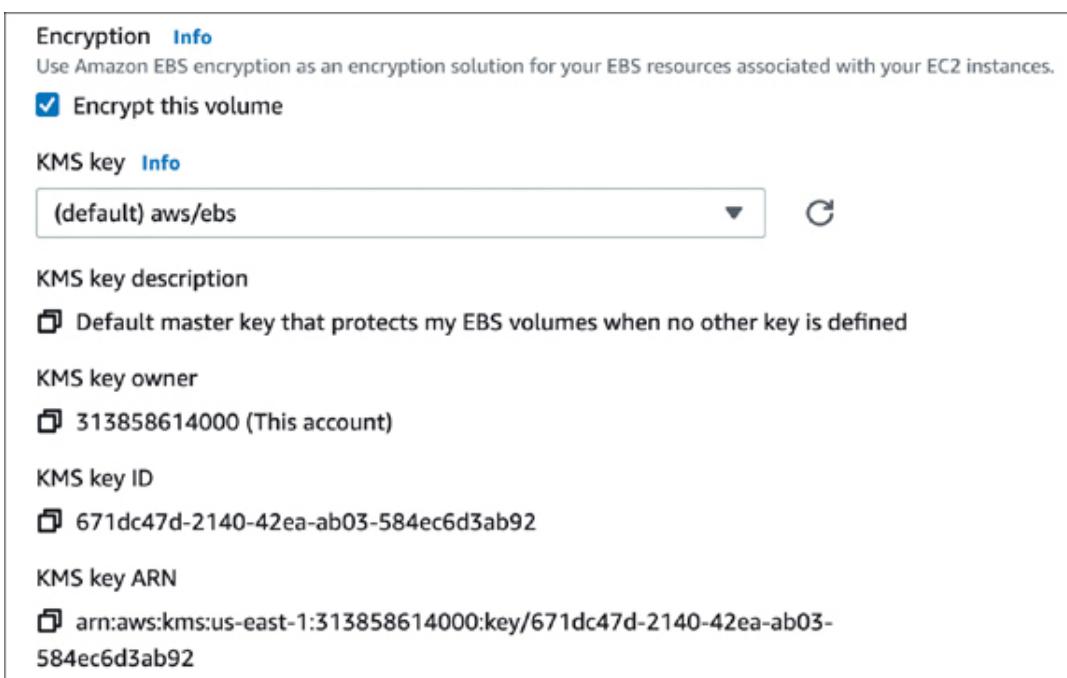
key administrator and to allow key rotation of your Customer Master Keys. Administrators can use the KMS master key provided to create additional AWS KMS administrators, and to optionally enable key rotation of the CMK (see [Figure 5-6](#)).

**Key Topic**

The screenshot shows the 'General configuration' tab of a KMS key named 'cloudtrail'. The key has an alias 'cloudtrail', is in an 'Enabled' status, and was created on Oct 06, 2020 at 23:03 EDT. It has a single ARN entry: arn:aws:kms:us-east-1:3138:58614000:key/bfe3c811-0430-4348-8432-8f4984293d78. The 'Description' field contains the text: 'The key created by CloudTrail to encrypt log files. Created Wed Oct 07 03:03:38 UTC 2020'. The 'Regionality' is listed as 'Single Region'. Below the general configuration, there are tabs for 'Key policy', 'Cryptographic configuration', 'Tags', 'Key rotation' (which is currently selected), and 'Aliases'. Under the 'Key rotation' tab, there is a section titled 'Key rotation' with a 'Save' button. A checkbox labeled 'Automatically rotate this KMS key every year.' is checked, with a link 'Learn more' next to it.

**Figure 5-6** Enabling Key Rotation

To encrypt an EBS volume using the AWS Key Management Service, a CMK can be created by AWS and stored in AWS KMS. Optionally, organizations can choose to specify the key material for the CMK, which can be generated by KMS or imported from your own key management infrastructure. After a CMK has been created, you can create an encrypted EBS volume using the EC2 dashboard and specifying the ID of the CMK when creating the volume (see [Figure 5-7](#)). The EBS volume will be encrypted using the specified CMK, and the data on the EBS volume will be encrypted at rest on the underlying storage.



**Figure 5-7** Select KMS Key

When you attach the encrypted EBS volume to an EC2 instance, the instance will automatically download and install the necessary encryption and decryption components, including the appropriate version of the AWS Encryption SDK and the public key portion of the CMK. The instance will then use the CMK to encrypt and decrypt data as it is written to and read from the EBS volume. The private key portion of the CMK remains securely stored in AWS KMS, and is never made available to the EC2 instance.

When an EBS volume has been encrypted and attached to an EC2 instance, the following data types are encrypted:

- Data at rest inside the EBS volume
- All data that moves between the attached EBS volume and the EC2 instance
- All snapshots created from the EBS volume
- All volumes created from the encrypted snapshots

AWS KMS performs the following steps, as illustrated in [Figure 5-8](#), to encrypt and decrypt the EBS volume:

**Step 1.** AWS EBS sends a request to KMS, specifying the CMK to use for the AWS EBS volume encryption.

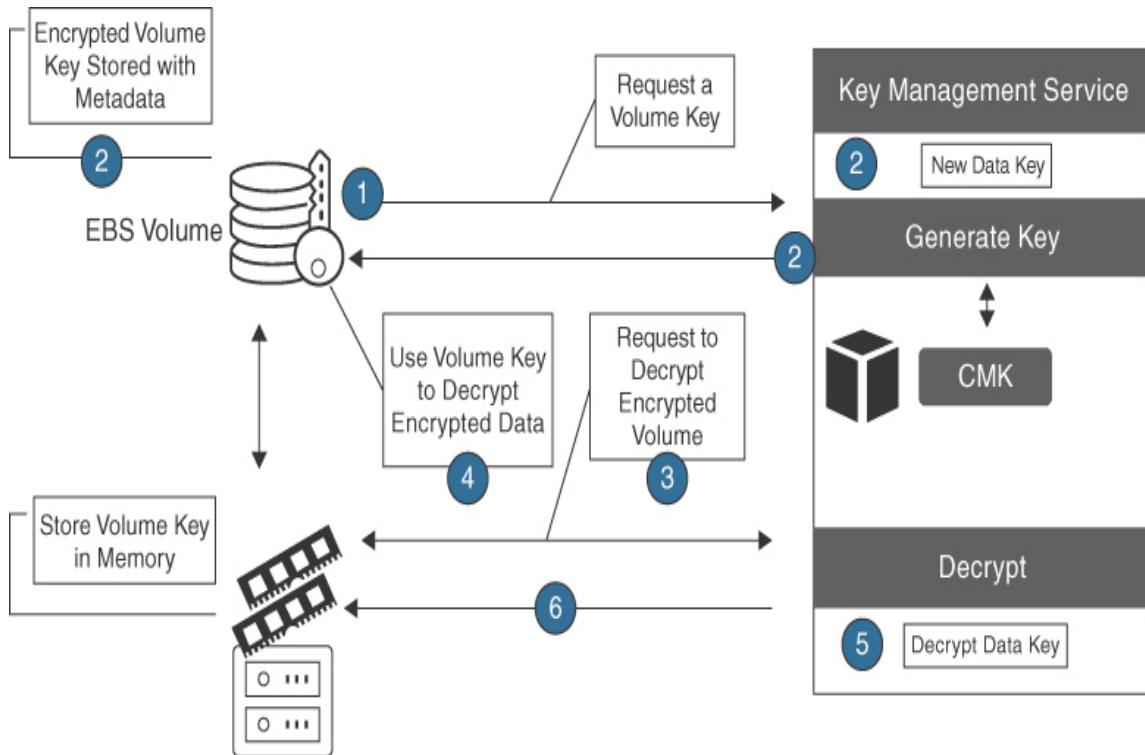
**Step 2.** AWS KMS generates a new data key, encrypts it using the specified CMK, and sends the encrypted key to AWS EBS to be stored with the volume metadata.

**Step 3.** The Amazon EC2 service sends a decrypt request to KMS.

**Step 4.** EBS sends a request to KMS to decrypt the data key.

**Step 5.** KMS uses the CMK to decrypt the encrypted data key and sends the decrypted key to the EC2 service.

**Step 6.** EC2 stores the plaintext decrypted key in protected hypervisor memory on the bare-metal server where the EC2 instance is hosted and uses the key when required to perform decryption for the EBS volume.



**Figure 5-8** EBS Encryption Steps

---

### Note

The default setting for each AWS region is that EBS encryption is not enabled. To enable EBS encryption in the AWS region, open the EC2 dashboard, and in the upper-right corner under Account Attributes click EBS Encryption. Click Manage and choose the desired AWS-managed CMK or another CMK. Next, click Enable and then click Update EBS encryption. Once encryption is

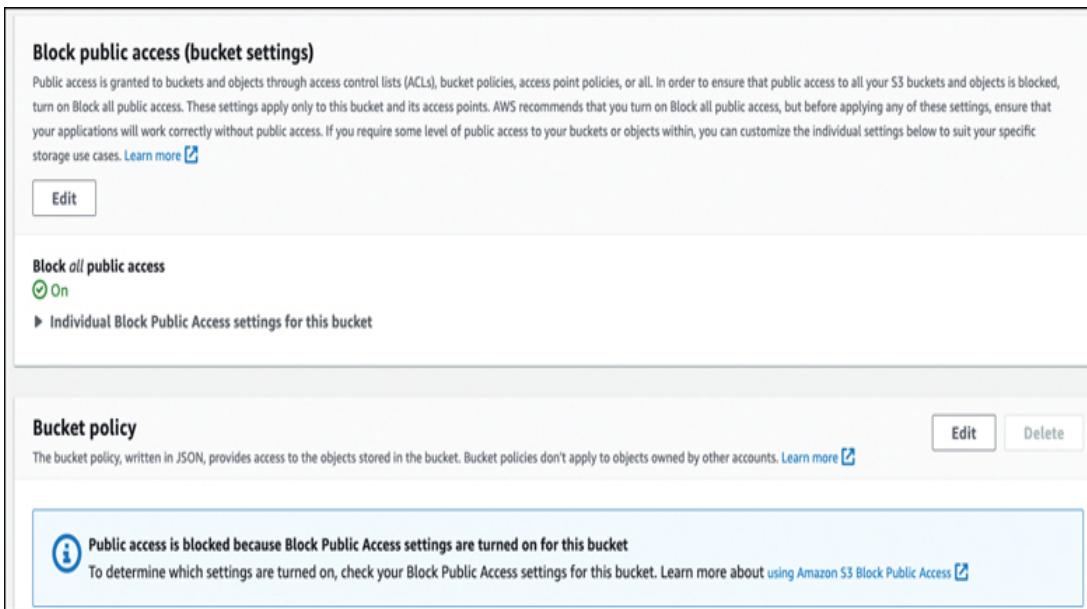
enabled for the AWS region, all new EBS volumes and snapshots will be encrypted at creation.

---

## Amazon S3 Bucket Security

By default, only the owner who created an S3 bucket has access to the objects stored in the bucket. There are several methods for controlling security for an S3 bucket (see [Figure 5-9](#)):

- **ACLs:** You can use [\*access control lists \(ACLs\)\*](#) to control primary access from other AWS accounts for list and write objects and read and write bucket permissions, public access, and access to S3 logging information. ACLs are available for purposes of backward compatibility and are the weakest type of S3 security (and therefore not recommended).



**Figure 5-9** S3 Permission Settings

- **IAM policy:** You can grant access to other AWS users and groups of IAM users by using IAM permission policies in partnership with resource policies.
- **S3 Bucket policy:** You can control direct access to an S3 bucket, as shown in [Example 5-2](#), by creating a **bucket policy** assigned directly to the S3 bucket. An S3 bucket policy is a JSON-formatted document that defines which actions are allowed or denied on an S3 bucket and its contents. A bucket policy is attached directly to the bucket it is protecting, and the policy settings list who has access to the bucket and what they can do with the objects in the bucket. An S3 bucket policy might allow a specific IAM user to read and write

objects in the bucket, while denying access to all other users. Or, the policy might allow any user to read objects in the bucket but allow only authenticated users to write objects. S3 bucket policies are defined using the AWS Policy Language, which provides a set of keywords and operations that you can use to specify the conditions under which a policy takes effect. A bucket policy can also allow access from multiple AWS accounts to a single S3 bucket.

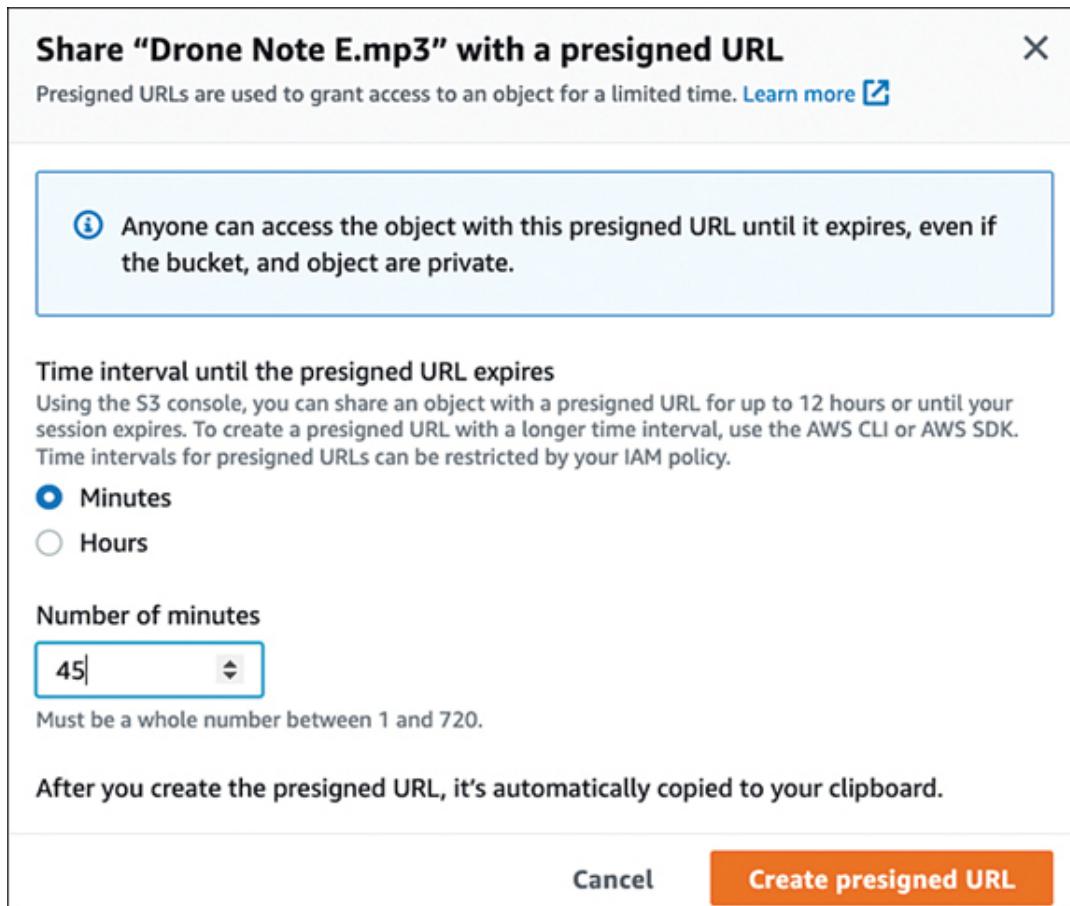
### Example 5-2 S3 Bucket Policy

[Click here to view code image](#)

```
{  
    "Version": "2012-10-17",  
    "Id": "S3PolicyId1",  
    "Statement": [  
        {  
            "Sid": "IPAllow",  
            "Effect": "Deny",  
            "Principal": "*",  
            "Action": "s3:*",  
            "Resource": [  
                "arn:aws:s3::::2021232reports",  
                "arn:aws:s3::::2021232reports/*"  
            ],  
        },  
    ]  
}
```

```
        "Condition": {
            "NotIpAddress": {"aws:SourceIp": "54.242.143.143"}
        }
    }
]
```

- **Query string authentication:** Query string authentication is a method to authenticate requests to an Amazon S3 bucket allowing organizations to generate a URL (see [Figure 5-10](#)) that can be shared with end users. When an end user clicks the URL, they are granted access to the specified S3 bucket and its contents.



**Figure 5-10** Presigned URL for S3 Object Access

The URL includes a set of parameters that specify the credentials that grant access to the bucket. These parameters include the access key ID, an expiration time for the URL, and a signature that is calculated using the access key secret. When someone attempts to access the URL, the Amazon S3 service checks the signature to verify that it matches the expected value. If the signature is valid, the user is granted access to the bucket; otherwise, the request is denied.

The use case for using query string authentication is useful for granting temporary access to an S3 bucket without having to create an IAM user or provide AWS access keys. However, query string authentication is not as secure as IAM policies or bucket policies because the URL and its parameters are included in each request; therefore, anyone who has access to the URL can potentially gain access to the bucket.

---

#### Note

If you require public access to objects in an S3 bucket, it's recommended that you create a separate AWS account specifically for hosting the S3 buckets that will have public S3 object access.

---

- **Blocking S3 public access:** S3 Buckets always start as private, with no default public access (see [Figure 5-11](#)). When the Block Public Access (Bucket Settings) setting is enabled, attempts at changing security settings to allow public access to objects in the S3 bucket are denied. You can block public access on an individual S3 bucket or on all S3 buckets in your AWS account by editing the public access settings for your account using the S3 console. Choices for blocking S3 public access include the following:

- **Public:** Everyone has access to list objects, write objects, and read and write permissions.
- **Objects Can Be Public:** The bucket is not public; however, public access can be granted to individual objects by users with permissions.
- **Buckets and Objects Not Public:** No public access is allowed to the bucket or the objects within the bucket.

**Block public access (bucket settings)**

Public access is granted to buckets and objects through access control lists (ACLs), bucket policies, access point policies, or all. In order to ensure that public access to all your S3 buckets and objects is blocked, turn on Block all public access. These settings apply only to this bucket and its access points. AWS recommends that you turn on Block all public access, but before applying any of these settings, ensure that your applications will work correctly without public access. If you require some level of public access to your buckets or objects within, you can customize the individual settings below to suit your specific storage use cases. [Learn more](#)

**Block all public access**

- On
  - Block public access to buckets and objects granted through *new* access control lists (ACLs)
  On
  - Block public access to buckets and objects granted through *any* access control lists (ACLs)
  On
  - Block public access to buckets and objects granted through *new* public bucket or access point policies
  On
  - Block public and cross-account access to buckets and objects through *any* public bucket or access point policies
  On

**Figure 5-11** Blocking Public Access on an S3 Bucket by Default

---

### Note

Amazon Macie is a powerful AWS security service that uses artificial intelligence (AI) and machine learning (ML) technology to analyze your S3

objects and access patterns. Amazon S3 data can be classified based on many file formats, such as Personally Identifiable Information (PII) and other file types. AWS SNS notifications can be generated by Amazon Macie when Amazon S3 objects are discovered to be compromised.

---

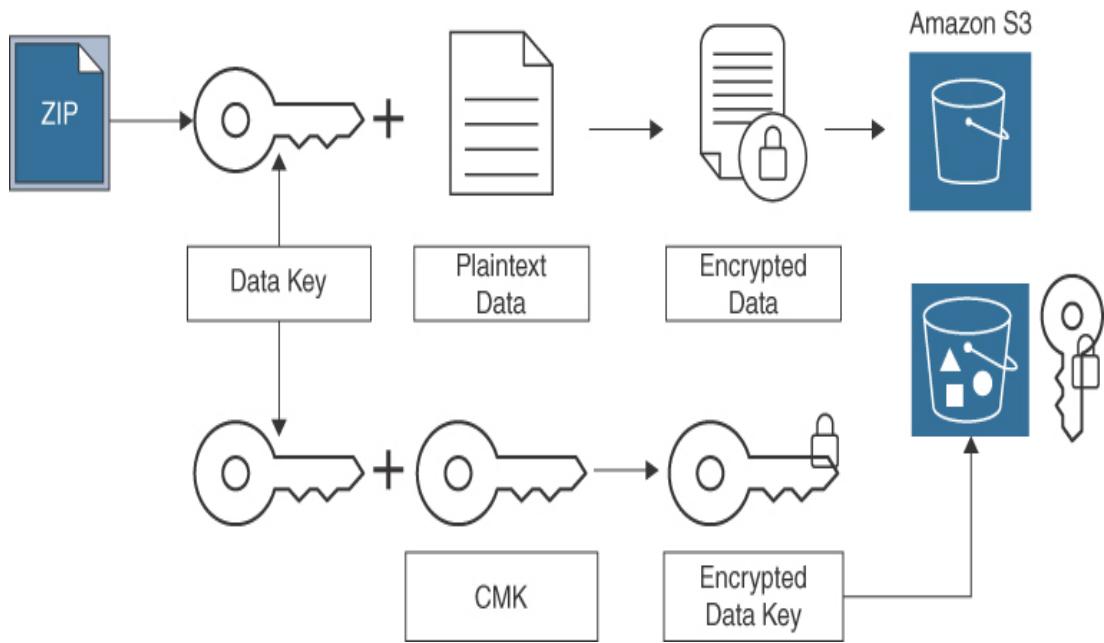
## S3 Storage at Rest



For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, the key topics to know about S3 storage at rest are as follows:

- **SSE-S3:** With SSE-S3, Amazon S3 manages the encryption and decryption of the data in the bucket. Organizations that select this option don't manage the encryption keys but can access the data in the bucket without having to manage the keys. SSE-S3 uses the Advanced Encryption Standard (AES) algorithm with a 256-bit key to encrypt the data in the bucket. The key is automatically generated by Amazon S3 and is regularly rotated to ensure the security of the encrypted data (see [Figure 5-12](#)). Note that SSE encrypts the

object data but the optional tag object metadata remains unencrypted.

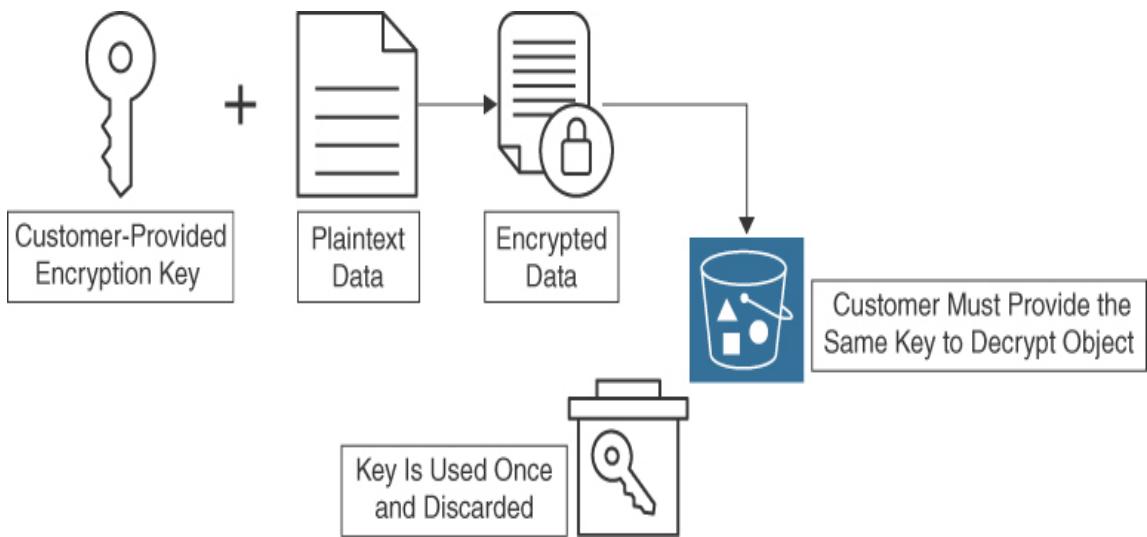


**Figure 5-12 SSE-S3 Encryption Process**

- **SSE-KMS:** Organizations can select AWS KMS to manage their encryption keys. Select the default CMK or choose a CMK that was already created in AWS KMS before starting an S3 encryption process. Accessing encrypted objects managed by KMS can be expensive: If you have an exceptionally large number of encrypted objects, a large volume of decryption requests will be made to KMS. You can configure SSE-KMS to significantly reduce the cost of the encryption and decryption process. When an S3 Bucket Key is configured for SSE-KMS

server-side encryption, a short-lived encryption key is created and stored and used to encrypt objects internally inside AWS S3 rather than utilize AWS KMS encryption processes. The S3 Bucket Key creates unique data keys for encrypting objects in the specific S3 bucket that has enabled the S3 Bucket Key option. The encryption process reduces AWS KMS requests for external encryption keys and can reduce encryption costs by 99%. The S3 Bucket Key is a worker process within the S3 bucket that enables you to perform encryption services without constant communication with KMS.

- **SSE-C:** You can use SSE with a customer-provided encryption key. With each request, the encryption key is provided to AWS, and Amazon S3 manages the encryption and decryption of S3 objects by using the supplied key. The same encryption key that was used to encrypt the object must be provided before the object can be decrypted (see [Figure 5-13](#)). After the encryption process is complete, the supplied encryption key is deleted from memory. To upload an object with an organization-provided encryption key (SSE-C), the AWS CLI, AWS SDK, or Amazon S3 REST API must be used.



**Figure 5-13 SSE-C Encryption Process**

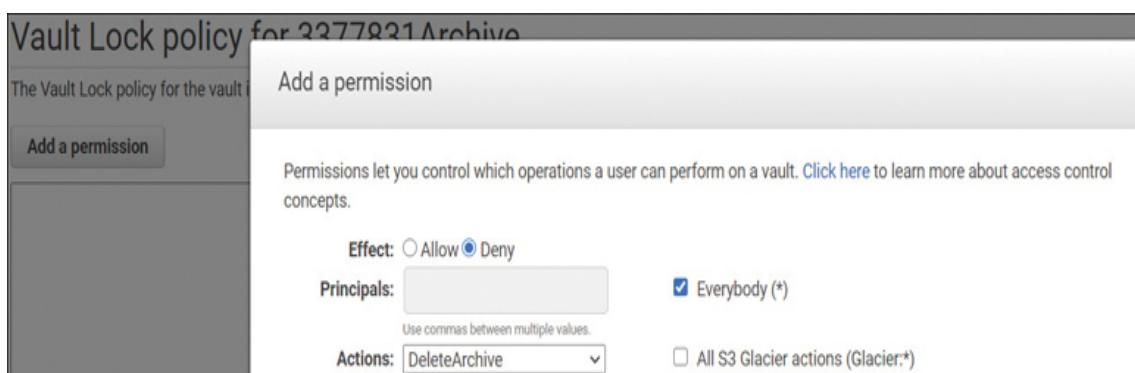
## Amazon S3 Object Lock Policies



Amazon S3 buckets and Amazon S3 Glacier have data policies that can lock objects so they cannot be deleted or changed. Amazon S3 objects can be locked using a [write-once/read-many \(WORM\)](#) policy. Object lock policies enable you to set rules that restrict certain actions on objects, such as deleting or overwriting them, in order to protect objects and ensure they remain available and unaltered. Object lock policies are set at the S3 bucket level and apply to all objects in the bucket, or set on individual objects. This can be useful for complying with

legal or regulatory requirements or protecting important or sensitive data. Apply a WORM policy, as shown in [Figure 5-14](#), to stop an Amazon S3 object from being overwritten, or deleted for a fixed time period, or indefinitely. There are several options to WORM policies to understand. First is the *retention period*, which refers to a set number of days or years during which an object will remain locked, protected, and unable to be overwritten or deleted. There are two retention modes:

- **Governance mode:** An S3 object cannot have its lock settings overwritten and cannot itself be overwritten or deleted unless the user has unique permissions. To override governance mode retention settings, an IAM user must have the **s3: BypassGovernanceRetention** permission and **x-amz-bypass-governance-retention: true** applied.
- **Compliance mode:** A protected object in your AWS account cannot be overwritten or deleted by anyone, including the root user, for the entire retention period.



**Figure 5-14** WORM Policy Settings

## Legal Hold

An object lock allows you to place a legal hold on an S3 object. Legal hold provides the same protection as a previously discussed retention period but does not have an expiration date. Once in force, a legal hold remains in place until it is removed. An object lock works on S3 buckets that have versioning already enabled. Legal hold can be applied to a single S3 object. A legal hold can be placed and removed by any user with the **s3:PutObjectLegalHold** permission applied to their IAM user or group account they are a member of.

---

### Note

Object lock can only be enabled for new buckets when they are being created.

---

## Amazon S3 Glacier Storage at Rest



Objects stored in Amazon S3 Glacier are automatically encrypted using SSE and AES-256 encryption. Amazon S3 Glacier Vault Lock enables you to deploy and enforce regulatory and required compliance controls by applying a Vault Lock policy on an Amazon S3 Glacier vault. Once a WORM policy has been applied to an S3 Glacier vault, the policy cannot be changed.

---

#### Note

Both EFS and FSx use AES-256 encryption to encrypt EFS data and metadata at rest. When your file system is mounted, you can also encrypt your EFS data in transit with TLS. FSx also supports the encryption of data in transit on file shares mapped on a computer instance that supports SMB Version 3.0 or newer. Encryption of data records at rest is automatically enabled when an FSx file system is created.

---

## Data Backup and Replication

Amazon S3 object backups can be carried out with the services and utilities listed in [Table 5-2](#). AWS Backup and AWS DataSync can back up additional AWS storage service data records.

**Table 5-2** Data Backup and Replication Options

AWS Service	Use	Data Types
AWS Backup	Back up all AWS storage services	EBS volumes and snapshots, S3 buckets, EFS, FSx for Windows File Server, RDS, DynamoDB
Amazon S3 Same-Region Replication (SRR)	Replicate objects to an S3 bucket in the same AWS region	Objects and versioned objects

AWS Service	Use	Data Types
-------------	-----	------------

Amazon S3 Cross-Region Replication (CRR)	Replicate objects to an S3 bucket in a different AWS region	Objects and versioned objects
--	---	-------------------------------

Amazon S3 Multi-Region Access Points	Replicate data sets across multiple AWS regions	Objects and versioned objects
--------------------------------------	---	-------------------------------

AWS  
Service

Use

Data Types

AWS DataSync AWS storage services Copy data to and from AWS storage services Network File System (NFS) or Server Message Block (SMB) shares, Hadoop Distributed File Systems (HDFS), AWS Snowcone, S3 buckets, EFS, FSx for Windows File Server

## AWS Key Management Service

Key  
Topic

**AWS Key Management Service (KMS)** lets organizations create, manage, and control cryptographic keys used to protect data records. AWS KMS integrates with AWS services that can encrypt data records (see [Figure 5-15](#)).

The screenshot shows the AWS KMS console interface. On the left, there is a sidebar with the title "Key Management Service (KMS)" and three navigation options: "AWS managed keys" (which is highlighted in orange), "Customer managed keys", and "Custom key stores". The main area is titled "AWS managed keys (13)". It includes a search bar with the placeholder "Filter keys by alias or key ID" and a pagination control showing page 1 of 2. A table lists 13 AWS managed keys, each with its alias, key ID, and status. The columns are "Aliases", "Key ID", and "Status".

Aliases	Key ID	Status
aws/lambda	4e348669-5704-4079-922c-0e6559a47794	Enabled
aws/acm	5e734f45-b808-4279-a782-948455960f32	Enabled
aws/ebs	671dc47d-2140-42ea-ab03-584ec6d3ab92	Enabled
aws/elasticfilesystem	763b4b16-998c-4a54-aee8-eca63bd53cee	Enabled
aws/cloud9	a12a8290-9390-4770-b537-b89fd6ecd52d	Enabled

**Figure 5-15** KMS Console

Organizations do not have to directly interface with AWS KMS to enable data encryption; instead, they can use AWS KMS services through more than 100 integrated AWS services, such as Amazon EBS storage, Amazon RDS, Amazon S3, Amazon EFS, Amazon FSx for Windows File Server, Amazon Aurora, and Amazon DynamoDB. When you enable encryption services using AWS KMS, a CMK is automatically generated in your AWS account for data encryption and decryption services.

Organizations can choose to create one or more CMKs and use them to match their security requirements. A custom CMK allows you to control each key's access control and usage policy; you can also grant permissions to other AWS accounts and services to use a specific custom CMK.

You can also choose to create symmetric CMKs, which use the same key to encrypt and decrypt data, or asymmetric CMKs, which use a public/private key pair (one for encrypting and one for decrypting).

The most common way to use KMS is to choose which AWS service will encrypt your data and select the CMK from within the AWS service itself; for example, you can encrypt an RDS database volume, as shown in [Figure 5-16](#).



**Figure 5-16** Generating CMKs with KMS for an RDS Instance

## Envelope Encryption

KMS uses a process called *envelope encryption* to encrypt data at rest. It involves two layers of encryption: the first layer encrypts the data using a key generated by the organization, and the

second layer encrypts the customer-generated key using a key that is managed by the AWS Key Management Service (KMS). This process enables each organization to retain control over their encryption keys and also enables them to rotate and manage the keys as needed, while still benefitting from the security and reliability of using the KMS for encryption key management. When you need to encrypt data, KMS generates a data key that is used to encrypt the data locally within the AWS service or application. The data keys are also encrypted under the organization's CMK. When it's time to decrypt your data, a request is sent to KMS to decrypt the data key (that is, the data key copy that was stored with the encrypted data) using your CMK. The entire encryption or decryption process is logged in AWS CloudTrail for auditing purposes.

---

#### Note

You can create up to 10,000 CMKs per AWS account per AWS region. Keys generated by AWS KMS can be enabled to be automatically rotated on an annual basis. However, automatic key rotation is not supported for external cryptographic keys imported into AWS KMS.

---

Organizations that choose to import 256-bit symmetric keys into AWS KMS for compliance requirements are responsible for managing the imported keys' expiration dates.

In addition to encrypting your data, AWS KMS provides other security features to help protect your encryption keys:

- **Key management:** As an administrator, you can create, rotate, disable, and delete the CMKs that are used to encrypt your data. You can also view the key policy for a CMK, which specifies who has access to the CMK and what actions they can perform with it.
- **Access control:** Organizations can use AWS IAM policies to control who has access to their CMKs and what actions can be performed with them. For example, users can be granted the ability to encrypt data using a specific CMK, but not to decrypt it or change the key policy.
- **Auditing:** AWS KMS logs all API calls to AWS CloudTrail so organizations can track who is using each CMK and for what purpose. Auditing can help ensure that encryption keys are being used securely and in accordance with an organization's security policies.
- **Key material:** KMS stores the key material for your CMKs in secure hardware devices called hardware security modules (HSMs). This helps protect the security of each organization's

keys and ensures that they are only accessible to authorized users.

- **Key rotation:** CMKS can be configured to automatically be rotated on an annual basis, to help prevent security breaches.

## AWS KMS Cheat Sheet



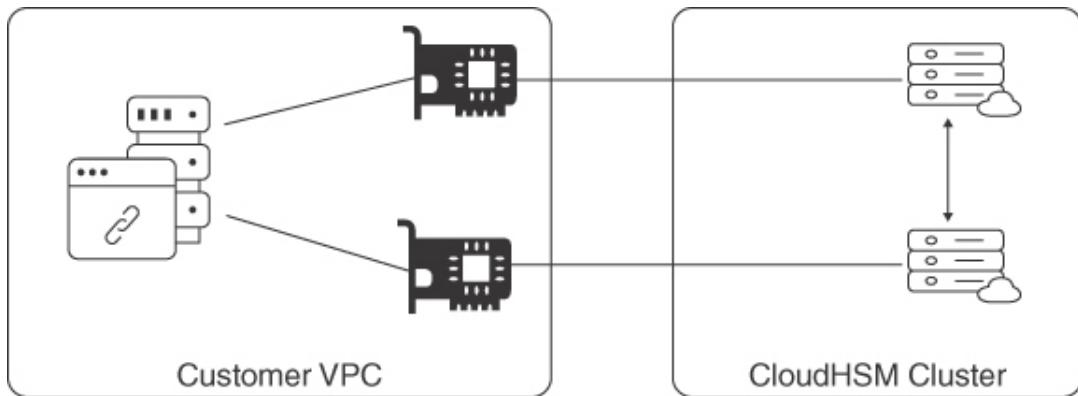
For the AWS Certified Solutions Architect – Associate (SAA-C03) exam, you need to understand the following critical aspects of AWS KMS:

- AWS KMS can be used to create symmetric keys within a custom key store such as AWS CloudHSM.
- An organization's symmetric keys can be imported for use with AWS KMS.
- AWS KMS can create symmetric and asymmetric data key pairs for application use.
- CMKs can be automatically rotated annually.
- CMKs can be disabled and re-enabled.
- AWS KMS keys can be audited with AWS CloudTrail.

## AWS CloudHSM



Instead of using the default AWS KMS store, you can create a custom key store using a VPC-hosted AWS CloudHSM cluster and authorize KMS to use it as its dedicated key store. AWS CloudHSM clusters are created using multiple single-tenant hardware devices (see [Figure 5-17](#)). Amazon maintains the AWS CloudHSM hardware and backs up its contents but never enters an AWS CloudHSM device. Organizations might use an AWS CloudHSM deployment if compliance rules explicitly require that encryption keys are protected in a single-tenant hardware device. AWS CloudHSM can operate as a complete stand-alone hardware device for your synchronous and asynchronous keys and provide you with Federal Information Processing Standard (FIPS) 140-2 Level 3 compliance.



**Figure 5-17** CloudHSM Design

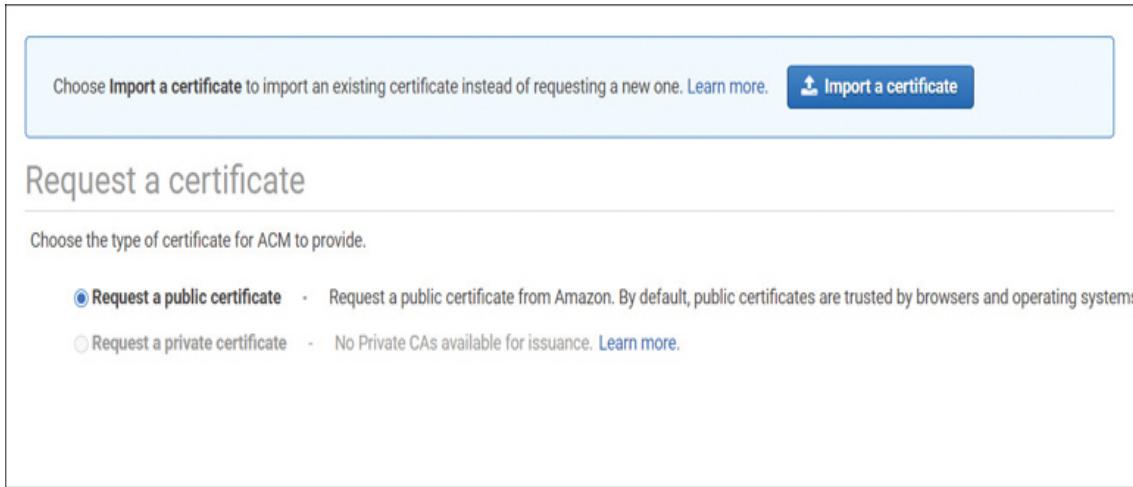
## AWS Certificate Manager

AWS Certificate Manager (ACM) is a managed service that allows you to provision, manage, and deploy public and private SSL/TLS certificates that can be used with your AWS services and AWS-hosted websites and applications. Certificates can also be deployed on ELB load balancers, CloudFront distributions, Elastic Beanstalk, and APIs hosted on Amazon API Gateway. There is no additional charge for provisioning public or private SSL/TLS certificates for use with AWS services. However, organizations will pay a fee for creating and operating a private ***certificate authority (CA)*** and for the private certificates that are issued by the private CA that is used by your internally hosted resources, such as application servers or appliances.

ACM can generate the following certificate types (see [Figure 5-18](#)):

**Key  
Topic**

- **Public certificates:** ELB port 443 traffic, CloudFront distributions, and public-facing APIs hosted by Amazon API Gateway all use public certificates. Use AWS Certificate Manager to request a public certificate for a domain name for your site. AWS Certificate Manager validates that you own or control the domain name in your certificate request. Validation options include DNS validation and email validation.
- **Private certificates:** Delegated private certificates are managed by an AWS Certificate Manager–hosted private CA, which can automatically renew and deploy certificates for private-facing Amazon ELB and Amazon API Gateway deployments. Private certificates can also secure Amazon EC2 instances, Amazon ECS containers, and IoT devices.
- **Imported certificates:** Third-party certificates can be imported into AWS Certificate Manager.
- **CA certificates:** Certificates can be issued for creating a private CA up to five levels deep, including a root CA, three levels of subordinate CAs, and a single issuing CA.



**Figure 5-18** Certificate Choices in AWS Certificate Manager

## Encryption in Transit

AWS uses HTTPS endpoints communication, providing encryption in transit for communicating with AWS APIs. AWS service endpoints can also be accessed using TLS version 1.2. Some AWS services offer endpoints that support the Federal Processing Standard (FIPS) 140-2 in some regions. Each endpoint is the URL of the entry point for each AWS service. AWS SDKs and the AWS Command Line Interface (AWS CLI) automatically use the default endpoint for each service per AWS Region, but an alternative endpoint can be specified for API requests. Most AWS services have regional endpoints that can be used to make requests. The format for a regional endpoint is *protocol://service-code.region-code.amazonaws.com*. AWS endpoints can be referenced here:

<https://docs.aws.amazon.com/general/latest/gr/aws-service-information.html>.

Global endpoints are used for global services and services located in edge locations. The global AWS services are

- Amazon CloudFront
- AWS Global Accelerator
- AWS Identity and Access Management (IAM)
- AWS Organizations
- Amazon Route 53
- AWS Shield Advanced
- AWS WAF Classic

HTTP endpoints for domains and hosted workloads hosted at AWS can be blocked with Security Groups and Network ACLs and can automatically be redirected to HTTPS endpoints when using Amazon CloudFront or an Amazon ELB.

## **Exam Preparation Tasks**

As mentioned in the section “[How to Use This Book](#)” in the Introduction, you have a couple of choices for exam preparation: the exercises here, [Chapter 16](#), “[Final Preparation](#),” and the exam simulation questions in the Pearson Test Prep software online.

## Review All Key Topics

Review the most important topics in the chapter, noted with the Key Topic icon in the margin of the page. [Table 5-3](#) lists these key topics and the page number on which each is found.



**Table 5-3** [Chapter 5](#) Key Topics

Key Topic Element	Description	Page Number
<a href="#">Figure 5-1</a>	Encryption Choices at AWS	204
Section	Data Retention and Classification	207
Section	Infrastructure Security	209
Section	Detective Controls	210
Section	Amazon EBS Encryption	212

Key Topic Element	Description	Page Number
<u>Figure 5-6</u>	Enabling Key Rotation	213
Section	S3 Storage at Rest	220
Section	Amazon S3 Object Lock Policies	221
Section	Amazon S3 Glacier Storage at Rest	222
Section	AWS Key Management Service	224
Section	AWS KMS Cheat Sheet	226
Section	AWS CloudHSM	227
List	AWS Certificate Manager certificate types	227

## Define Key Terms

Define the following key terms from this chapter and check your answers in the glossary:

Amazon Elastic Block Storage (EBS)

symmetric key

access control list (ACL)

bucket policy

write-once/read-many (WORM)

AWS Key Management Service (KMS)

certificate authority (CA)

## Q&A

The answers to these questions appear in Appendix A. Use the Pearson Test Prep Software Online for more practice with exam format questions.

1. Which AWS storage service is available with AWS as a single-tenant storage design?

- 2.** What is the default state of an S3 bucket regarding public access when the bucket is first created?
- 3.** What is the security advantage of using SSE-C encryption with Amazon S3 buckets?
- 4.** Describe the concept of envelope encryption that KMS uses.
- 5.** What type of data stored at AWS is always automatically encrypted by default?
- 6.** Why is AWS CloudHSM chosen by companies that must adhere to a high compliance standard?
- 7.** How does AWS KMS carry out automatic key rotation for imported keys?
- 8.** Where can private CAs created by AWS Certificate Manager be deployed?