AWS Startup Security Baseline

# AWS Prescriptive Guidance

# AWS Prescriptive Guidance: AWS Startup Security Baseline

# Table of Contents

# AWS Startup Security Baseline

*Amazon Web Services* (contributors)

*May 2023* (document history)

The AWS Startup Security Baseline (AWS SSB) is a set of controls that create a minimum foundation for businesses to build securely on AWS without decreasing their agility. These controls form the basis of your security posture and are focused on securing credentials, enabling logging and visibility, managing contact information, and implementing basic data boundaries.

The controls in this guide are designed with early startups in mind, mitigating the most common security risks without requiring significant effort. Many startups begin their journey in the AWS Cloud with a single AWS account. As organizations grow, they migrate to multi-account architectures. The guidance in this guide is designed for single-account architectures, but it helps you set up security controls that are easily migrated or modified as you transition to a multi-account architecture.

The controls in the AWS SSB are separated into two categories: account and workload. Account controls help keep your AWS account secure. It includes recommendations for setting up user access, policies, and permissions, and it includes recommendations for how to monitor your account for unauthorized or potentially malicious activity. Workload controls help secure your resources and code in the cloud, such as applications, backend processes, and data. It includes recommendations such as encryption and reducing the scope of access.

> **Note**
>
> Some of the controls recommended in this guide replace the defaults configured during initial setup, while most configure new settings and policies. This document should in no way be considered comprehensive of all available controls.

## Intended audience

This guide is best suited for startups that are in the very beginning stages of development, with minimal staff and operations.

Startups or other businesses that are in later stages of operation and growth can still derive significant value from reviewing these controls against their current practices. If you identify

any gaps, you can implement the individual controls in this guide and then evaluate them for appropriateness as a long-term solution.

> **Note**
>
> The recommended controls in this guide are foundational in nature. Startups or other companies operating at a later stage of scale or sophistication should add additional controls as applicable.

## Foundational framework and security responsibilities

AWS Well-Architected helps cloud architects build a secure, high-performing, resilient, and efficient infrastructure for their applications and workloads. The AWS Startup Security Baseline aligns to the security pillar of the AWS Well-Architected Framework. The *security pillar* describes how to take advantage of cloud technologies to protect data, systems, and assets in a way that can improve your security posture. This helps you meet your business and regulatory requirements by following current AWS recommendations.

You can assess your adherence to Well-Architected best practices by using the AWS Well-Architected Tool in your AWS account.

Security and compliance are a shared responsibility between AWS and the customer. The shared responsibility model is often described by saying that AWS is responsible for the security of the cloud (that is, for protecting the infrastructure that runs all the services offered in the AWS Cloud), and you are responsible for the security in the cloud (as determined by the AWS Cloud services that you select). In the shared responsibility model, implementing the security controls in this document is part of your responsibility as a customer.

# Securing your account

Controls and recommendations in this section help keep your AWS account secure. It emphasizes using AWS Identity and Access Management (IAM) users, user groups, and roles (also known as *principals*) for both human and machine access, restricting the use of the root user, and requiring multi-factor authentication. In this section, you confirm that AWS has the contact information necessary to reach you regarding your account activity and status. You also set up monitoring services, such as AWS Trusted Advisor, Amazon GuardDuty, and AWS Budgets, so that you are notified of activity in your account and can respond quickly if the activity is unauthorized or unexpected.

**This section contains the following topics:**



# ACCT.01 Set account-level contacts to valid email distribution lists

When setting up primary and alternate contacts for your AWS account, use an email distribution list instead of an individual's email address. Using an email distribution list makes sure that ownership and reachability are preserved as individuals in your organization come and go. Set alternate contacts for billing, operations, and security notifications, and use appropriate email

distribution lists accordingly. AWS uses these email addresses to contact you, so it is important you retain access to them.

**To edit your account name, root user password, or root user email address**

1. Sign in to the **Account Settings** page in the Billing and Cost Management console.
2. On the **Account Settings** page, next to **Account Settings**, choose **Edit**.
3. Next to the field you want to update, choose **Edit**.
4. After you have entered your changes, choose **Save changes**.
5. After you have made all of your changes, choose **Done**.

**To edit your contact information**

1. On the Account Settings page, under **Contact Information**, choose **Edit**.
2. For the fields you want to change, enter your updated information, and then choose **Update**.

**To add, update, or remove alternate contacts**

1. On the Account Settings page, under **Alternate Contacts**, choose **Edit**.
2. For the fields you want to change, enter your updated information, and then choose **Update**.

## ACCT.02 Restrict use of the root user

The root user is created when you sign up for an AWS account, and this user has full ownership privileges and permissions over the account that cannot be changed. Only use the root user for the specific tasks that require it. For more information, see Tasks that require root user credentials (IAM documentation). Perform all other actions in your account by using other types of IAM identities, such as federated users with IAM roles. For more information, see AWS security credentials (IAM documentation).

**To restrict use of the root user**

1. Require multi-factor authentication (MFA) for the root user as described in ACCT.05 Require multi-factor authentication to log in.
2. Create an administrative user so that you don't use the root user for everyday tasks. For more information about configuring user access, see ACCT.03 Configure console access for each user.

# ACCT.03 Configure console access for each user

As a best practice, AWS recommends using temporary credentials to grant access to AWS accounts and resources. *Temporary credentials* have a limited lifetime, so you do not have to rotate them or explicitly revoke them when they're no longer needed. For more information, see Temporary security credentials (IAM documentation).

For human users, AWS recommends using federated identities from a centralized identity provider (IdP), such as AWS IAM Identity Center, Okta, Active Directory, or Ping Identity. Federating users allows you to define identities in a single, central location, and users can securely authenticate to multiple applications and websites, including AWS, by using just one set of credentials. For more information, see Identity federation in AWS and IAM Identity Center (AWS website).

> **Note**
>
> Identity federation can complicate the transition from a single-account architecture to a multi-account architecture. It is common for startups to delay implementing identity federation until they have established a multi-account architecture managed in AWS Organizations.

**To set up identity federation**

1. If you are using IAM Identity Center, see Getting started (IAM Identity Center documentation).

   If you are using an external or third-party IdP, see Identity providers and federation (IAM documentation).
2. Make sure that your IdP enforces multi-factor authentication (MFA).
3. Apply permissions according to ACCT.04 Assign permissions.

For startups that are not prepared to configure identity federation, you can create users directly in IAM. This is not a recommended security best practice because these are long-term credentials that never expire. However, this is a common practice for startups in early operation to prevent difficulty with transitioning to a multi-account architecture when they're operationally ready.

As a baseline, you can create an IAM user for each person who needs to access the AWS Management Console. If you configure IAM users, do not share credentials across users, and regularly rotate the long-term credentials.

> **Warning**
>
> IAM users have long-term credentials, which presents a security risk. To help mitigate this risk, we recommend that you provide these users with only the permissions they require to perform the task and that you remove these users when they are no longer needed.

**To create an IAM user**

1. Create IAM users (IAM documentation).
2. Apply permissions according to ACCT.04 Assign permissions.

## ACCT.04 Assign permissions

Configure user permissions in the account by assigning *policies* to their IAM identity (user group or role). You can customize the permissions, or you can attach AWS managed policies, which are standalone policies designed by AWS to provide permissions for many common use cases. If you customize permissions, follow the security best practice of granting least privilege. *Least privilege* is the practice of granting the minimum set of permissions that each user needs to perform their tasks.

If you are using federated identities, users access the account by assuming an IAM role through the external identity provider. The IAM role defines what users authenticated by your organization's IdP are allowed to do in AWS. You apply custom or AWS managed policies to this role to configure permissions.

**To assign permissions for federated identities**

- If you are using IAM Identity Center, see Use IAM policies in permission sets (IAM Identity Center documentation).

  If you are using an external or third-party IdP, see Adding IAM identity permissions (IAM documentation).

If you are using IAM users, you can use user groups or roles to manage permissions for multiple IAM users. We recommend user groups for startups because they are easier to manage and less prone to misconfiguration that could pose security risks for your account. Assign users to user groups based on their job functions. Examples of user groups include application, data, networking,

and Development Operations (DevOps) engineers. You can also divide the user types into smaller user groups based on decision-making authority, such as for senior or non-senior engineers.

**To assign permissions for IAM users**

1. Create IAM user groups (IAM documentation).
2. Attach an AWS managed policy to an IAM user group (IAM documentation).

# ACCT.05 Require multi-factor authentication to log in

With multi-factor authentication (MFA), users have a device that generates a response to an authentication challenge. Each user's credentials and device-generated response are required to complete the sign-in process. As a security best practice, enable MFA for AWS account access, especially for long-term credentials such as the account root user and IAM users.

**To set up MFA for the root user**

1. Sign in to the AWS Management Console.
2. On the right side of the navigation bar, choose your account name, and then choose **My Security Credentials**.
3. If necessary, choose **Continue to Security Credentials**.
4. Expand the **Multi-Factor Authentication (MFA)** section.
5. Choose **Activate MFA**.
6. Follow the wizard instructions to configure your MFA devices accordingly. For more information, see AWS Multi-factor authentication in IAM (IAM documentation).

**To set up MFA in IAM Identity Center**

- Enable MFA (IAM Identity Center documentation)

**To set up MFA for your own IAM user**

1. Using your sign-in credentials, sign in to the IAM console.
2. In the navigation bar on the upper right, choose your user name, and then choose **My Security Credentials**.

3. On the **AWS IAM credentials** tab, in the **Multi-factor authentication** section, choose **Manage MFA device**.

**To set up MFA for other IAM users**

1. Sign in to the AWS Management Console and open the IAM console.
2. In the navigation pane, choose **Users**.
3. Choose the name of the user for whom you want to enable MFA, and then choose the **Security credentials** tab.
4. Next to **Assigned MFA device**, choose **Manage**.
5. Follow the wizard instructions to configure your MFA devices accordingly. For more information, see AWS Multi-factor authentication in IAM (IAM documentation).

# ACCT.06 Enforce a password policy

Users log in to the AWS Management Console by providing sign-in credentials, and MFA is recommended. Require that passwords adhere to a strong password policy to help prevent discovery through brute force or social engineering.

For more information about the latest recommendations for strong passwords, see Password Policy Guide on the Center for Internet Security (CIS) website.

For IAM users, you can configure password requirements in a custom IAM password policy. For more information, see Setting an account password policy (IAM documentation).

**To create a custom password policy**

1. Sign in to the AWS Management Console and open the IAM console.
2. In the navigation pane, choose **Account settings**.
3. In the **Password policy** section, choose **Change password policy**.
4. Select the options that you want to apply to your password policy, and then choose **Save changes**.

# ACCT.07 Deliver CloudTrail logs to a protected S3 bucket

Actions taken by users, roles, and services in your AWS account are recorded as events in AWS CloudTrail. CloudTrail is enabled by default, and in the CloudTrail console, you can access 90 days of event history information. To view, search, download, archive, analyze, and respond to account activity across your AWS infrastructure, see Viewing events with CloudTrail Event history (CloudTrail documentation).

To retain CloudTrail history beyond 90 days with additional data, you create a new trail that delivers log files to an Amazon Simple Storage Service (Amazon S3) bucket for all event types. When you create a trail in the CloudTrail console, you create a multi-region trail.

**To create a trail that delivers logs for all AWS Regions to an S3 bucket**

1. Create a trail (CloudTrail documentation). On the **Choose log events** page, do the following:
    a. For **API activity**, choose both **Read** and **Write**.
    b. For preproduction environments, choose **Exclude AWS KMS events**. This excludes all AWS Key Management Service (AWS KMS) events from your trail. AWS KMS **read** actions such as `Encrypt`, `Decrypt`, and `GenerateDataKey` can generate a large volume of events.

       For production environments, choose to log **Write** management events, and clear the check box for **Exclude AWS KMS events**. This excludes high-volume AWS KMS read events but still logs relevant write events, such as `Disable`, `Delete`, and `ScheduleKey`. These are the minimum recommended AWS KMS logging settings for a production environment.
2. The new trail appears on the **Trails** page. In about 15 minutes, CloudTrail publishes log files that show the AWS application programming interface (API) calls made in your account. You can see the log files in the S3 bucket that you specified.

**To help secure the S3 buckets where you store CloudTrail log files**

1. Review the Amazon S3 bucket policy (CloudTrail documentation) for any and all buckets where you store log files and adjust it as needed to remove any unnecessary access.
2. As a security best practice, be sure to manually add an `aws:SourceArn` condition key to the bucket policy. For more information, see Create or update an Amazon S3 bucket to use to store the log files for an organization trail (CloudTrail documentation).
3. Enable MFA Delete (Amazon S3 documentation).

# ACCT.08 Prevent public access to private S3 buckets

By default, only the root user of the AWS account and the IAM principal, if used, have permissions to read and write to Amazon S3 buckets created by that principal. Additional IAM principals are granted access by using identity-based policies, and access conditions can be enforced by using a bucket policy. You can create bucket policies that grant the general public access to the bucket, a *public* bucket.

Buckets created on or after April 28, 2023 have the **Block Public Access** setting enabled by default. For buckets created before this date, users might misconfigure the bucket policy and unintentionally grant access to the public. You can prevent this misconfiguration by enabling the **Block Public Access** setting for each bucket. If you have no current or future use cases for a public S3 bucket, enable this setting at the AWS account level. This setting prevents policies that allow public access.

**To prevent public access to S3 buckets**

- Configure block public access settings for your S3 buckets (Amazon S3 documentation).

AWS Trusted Advisor generates a yellow finding for S3 buckets that allow list or read access to the public and generates a red finding for buckets that allow public uploads or deletes. As a baseline, follow the control ACCT.12 Monitor for and resolve high-risk issues by using Trusted Advisor to identify and correct misconfigured buckets. Publicly accessible S3 buckets are also indicated in the Amazon S3 console.

# ACCT.09 Delete unused VPCs, subnets, and security groups

To reduce the opportunity for security issues, delete or turn off any resources that are not being used. In a new AWS account, by default a virtual private cloud (VPC) is created automatically in every AWS Region, which enables you to assign public IP addresses in public subnets. However, if these VPCs are not needed, this introduces risk of unintended exposure of resources.

If they are not in use, delete the default VPCs in all Regions, not just those in the Regions where you might deploy workloads. Deleting a VPC also deletes its components, such as subnets and security groups.

> **Note**
>
> You can view all Regions and VPCs on the Amazon EC2 Global View console. For more information, see List and filter resources across Regions using Amazon EC2 Global View (Amazon EC2 documentation).

**To delete unused default VPCs**

1. Delete your VPC (Amazon VPC documentation).
2. Repeat as needed for VPCs in other Regions.

## ACCT.10 Configure AWS Budgets to monitor your spending

AWS Budgets enable monitoring of monthly costs and usage with notifications when costs are forecasted to exceed target thresholds. Forecasted cost notifications can provide an indication of unexpected activity, providing extra defense in addition to other monitoring systems, such as AWS Trusted Advisor and Amazon GuardDuty. Monitoring and understanding your AWS costs is also part of good operational hygiene.

**To set up a budget in AWS Budgets**

- Create a cost budget (AWS Budgets documentation).

## ACCT.11 Enable and respond to GuardDuty notifications

Amazon GuardDuty is a threat-detection service that continuously monitors for malicious or unauthorized behavior to help protect your AWS accounts, workloads, and data. When it detects unexpected and potentially malicious activity, GuardDuty delivers detailed security findings for visibility and remediation. GuardDuty can detect threats such as cryptocurrency mining activity, access from Tor clients and relays, unexpected behavior, and compromised IAM credentials. Enable GuardDuty and respond to findings to stop potentially malicious or unauthorized behavior in your AWS environment. For more information about findings in GuardDuty, see Finding types (GuardDuty documentation).

You can use Amazon CloudWatch Events to set up automated notifications when GuardDuty creates a finding or the finding changes. First, you set up an Amazon Simple Notification Service

(Amazon SNS) topic and add endpoints, or email addresses, to the topic. Then, you set up a CloudWatch event for GuardDuty findings, and the event rule notifies the endpoints in the Amazon SNS topic.

**To enable GuardDuty and GuardDuty notifications**

1. Enable Amazon GuardDuty (GuardDuty documentation).
2. Create a CloudWatch Events rule to notify you of GuardDuty findings (GuardDuty documentation).

# ACCT.12 Monitor for and resolve high-risk issues by using Trusted Advisor

AWS Trusted Advisor passively scans your AWS infrastructure for high-risk or high-impact issues related to security, performance, cost, and reliability. It provides detailed information about affected resources and remediation recommendations. For a complete list of checks and descriptions, see AWS Trusted Advisor check reference (Trusted Advisor documentation).

Review Trusted Advisor findings on a recurring basis, and remediate issues as necessary. If you have the AWS Business Support or Enterprise Support plans, you can subscribe to a weekly findings email. For more information, see Set up notification preferences (AWS Support documentation).

**To view issues in Trusted Advisor**

- Review each check category according to the instructions in View check categories (Support documentation). At a minimum, we recommend reviewing the **action recommended** issues, which are red.

# Securing your workloads

Controls and recommendations in this section help you secure your workloads running in AWS, while you are building them. They emphasize secure practices for managing application secrets and scope of access, minimizing access routes to private resources, and using encryption to protect data in transit and at rest.

**This section contains the following topics:**



## WKLD.01 Use IAM roles for compute environment permissions

In AWS Identity and Access Management (IAM), a *role* represents a set of permissions that can be assumed by a person or service for a configurable period of time. Using roles eliminates the need to store or manage long-term credentials, significantly reducing the chance of unintended use. Assign an IAM role directly to Amazon Elastic Compute Cloud (Amazon EC2) instances, AWS Fargate tasks and services, AWS Lambda functions, and other AWS compute services whenever supported. Applications that use an AWS SDK and run in these compute environments automatically use the IAM role credentials for authentication.

The approach and instructions for using IAM roles for each service can be found in the AWS Documentation for the service. For example, see the following:

- IAM roles for Amazon EC2 (Amazon EC2 documentation)
- IAM roles for tasks (Amazon Elastic Container Service documentation)
- Lambda execution role (Lambda documentation)

## WKLD.02 Restrict credential usage scope with resource-based policies permissions

Policies are objects that can define permissions or specify access conditions. There are two primary types of policies:

- *Identity-based policies* are attached to principals and define what the principal's permissions in the AWS environment.
- *Resource-based policies* are attached to a resource, such as an Amazon Simple Storage Service (Amazon S3) bucket, or virtual private cloud (VPC) endpoint. These policies specify which principals are allowed access, supported actions, and any other conditions that must be met.

For a principal to be allowed access to perform an action against a resource, it must have permission granted in its identity-based policy and meet the conditions of the resource-based policy. For more information, see Identity-based policies and resource-based policies (IAM documentation).

Recommended conditions for resource-based policies include:

- Restrict access to only principals in a specified organization (defined in AWS Organizations) by using the `aws:PrincipalOrgID` condition.
- Restrict access to traffic that originates from a specific VPC or VPC endpoint by using the `aws:SourceVpc` or `aws:SourceVpce` condition, respectively.
- Allow or deny traffic based on the source IP address by using an `aws:SourceIp` condition.

The following is an example of a resource-based policy that uses the `aws:PrincipalOrgID` condition to allow only principals in the **<o-xxxxxxxxxxx>** organization to access the **<bucket-name>** S3 bucket:

```
{
   "Version":"2012-10-17",
   "Statement":[
     {
       "Sid":"AllowFromOrganization",
       "Effect":"Allow",
       "Principal":"*",
       "Action":"s3:*",
       "Resource":"arn:aws:s3:::<bucket-name>/*",
       "Condition": {
         "StringEquals": {"aws:PrincipalOrgID":"<o-xxxxxxxxxxx>"}
       }
     }
   ]
}
```

## WKLD.03 Use ephemeral secrets or a secrets-management service

Application secrets consist largely of credentials, such as key pairs, access tokens, digital certificates, and sign-in credentials. The application uses these secrets to gain access to other services it depends upon, such as a database. To help protect these secrets, we recommend they are either ephemeral (generated at the time of request and short-lived, such as with IAM roles) or retrieved from a secrets-management service. This prevents accidental exposure through less secure mechanisms, such as persisting in static configuration files. This also makes it easier to promote application code from development to production environments.

For a secrets-management service, we recommend using a combination of Parameter Store, a capability of AWS Systems Manager, and AWS Secrets Manager:

- Use Parameter Store to manage secrets and other parameters that are individual key-value pairs, string-based, short in overall length, and accessed frequently. You use an AWS Key Management Service (AWS KMS) key to encrypt the secret. There is no charge to store parameters in the standard tier of Parameter Store. For more information about parameter tiers, see Managing parameter tiers (Systems Manager documentation).
- Use Secrets Manager to store secrets that are in document form (such as multiple, related key-value pairs), are larger than 4 KB (such as digital certificates), or would benefit from automated rotation.

You can use Parameter Store APIs to retrieve secrets stored in Secrets Manager. This allows you to standardize the code in your application when using a combination of both services.

**To manage secrets in Parameter Store**

1. Create a symmetric AWS KMS key (AWS KMS documentation).
2. Create a SecureString parameter (Systems Manager documentation). Secrets in Parameter Store use the `SecureString` data type.
3. In your application, retrieve a parameter from Parameter Store by using the AWS SDK for your programming language. For code examples, see GetParameter (AWS SDK Code Library).

**To manage secrets in Secrets Manager**

1. Create a secret (Secrets Manager documentation).
2. Retrieve secrets from AWS Secrets Manager in code (Secrets Manager documentation).

   It is important to read Use AWS Secrets Manager client-side caching libraries to improve the availability and latency of using your secrets (AWS blog post). Using client-side SDKs, which already have best practices implemented, should accelerate and simplify the use and integration of Secrets Manager.

## WKLD.04 Prevent application secrets from being exposed

During local development, application secrets can be stored in local configuration or code files and accidentally checked-in to source code repositories. Unsecured repositories hosted on public service providers can be subject to unauthorized access and subsequent discovery of these secrets. Use available tools to prevent secrets from being checked in. Incorporate checks for exposed secrets as part of your manual code review processes.

Some common tools that can prevent application secrets from being checked-in to source code repositories are:

- Gitleaks (GitHub repository)
- Whispers (GitHub repository)
- detect-secrets (GitHub repository)
- git-secrets (GitHub repository)

- TruffleHog (GitHub repository)

## WKLD.05 Detect and remediate exposed secrets

In WKLD.03 Use ephemeral secrets or a secrets-management service and WKLD.04 Prevent application secrets from being exposed, you put measures in place to protect secrets. In this control, you deploy a solution that can detect if secrets have bypassed these prevention measures, and you can remediate accordingly.

Amazon CodeGuru Reviewer detects application secrets in source code and provides a mechanism to remediate and publish the detected secrets in Secrets Manager. Application code for retrieving the secret from Secrets Manager is also provided. Conduct a cost-benefit analysis to determine if this solution is right for your business. As an alternative, some of the open-source solutions in WKLD.04 Prevent application secrets from being exposed provide detection capability for existing secrets.

**To set up CodeGuru Reviewer integration with Secrets Manager**

- Use CodeGuru Reviewer to identify hardcoded secrets and AWS Secrets Manager to secure them (AWS blog post and guided walkthrough).

## WKLD.06 Use Systems Manager instead of SSH or RDP

*Public subnets*, which have a default route pointing to an internet gateway, are inherently a greater security risk than *private subnets*, those with no route to the internet. You can run EC2 instances in private subnets and use the Session Manager capability of AWS Systems Manager to remotely access the instances through either the AWS Command Line Interface (AWS CLI) or AWS Management Console. You can then use the AWS CLI or console to start a session that connects into the instance through a secure tunnel, preventing the need to manage additional credentials used for Secure Shell (SSH) or Windows remote desktop protocol (RDP).

Use Session Manager instead of running EC2 instances in public subnets, running jump boxes, or running bastion hosts.

**To set up Session Manager**

1. Make sure the EC2 instance is using the latest operating system Amazon Machine Images (AMIs), such as Amazon Linux or Ubuntu. The AWS Systems Manager Agent (SSM Agent) is pre-installed on the AMI.
2. Make sure the instance has connectivity, either through an internet gateway or through VPC endpoints, to these addresses (replacing **<Region>** with the appropriate AWS Region):
    a. `ec2messages.<Region>.amazonaws.com`
    b. `ssm.<Region>.amazonaws.com`
    c. `ssmmessages.<Region>.amazonaws.com`
3. Attach the AWS managed policy `AmazonSSMManagedInstanceCore` to the IAM role that is associated to your instances.

For more information, see Setting up Session Manager (Systems Manager documentation).

**To start a session**

- Start a session (Systems Manager documentation).

## WKLD.07 Log data events for S3 buckets with sensitive data

By default, AWS CloudTrail captures management events, events that create, modify, or delete resources in your account. These management events do not capture read or write operations to individual objects in Amazon Simple Storage Service buckets. During a security event, it is important to capture unauthorized data access or use at an individual record or object level. Use CloudTrail to log data events for any S3 buckets that store sensitive or business-critical data, for detection and auditing purposes.

> **Note**
>
> Additional charges apply for logging data events. For more information, see AWS CloudTrail pricing.

**To log data events for trails**

1. Sign in to the AWS Management Console and open the CloudTrail console
2. In the navigation pane, choose **Trails**, and then choose a trail name.
3. In **General details**, choose Edit to change the following settings. You cannot change the name of a trail.
    a. In **Data events**, choose **Edit**.
    b. For **Data event source**, choose **S3**.
    c. For **All current and future S3 buckets**, clear **Read** and **Write**.
    d. In Individual bucket selection, browse for the bucket on which to log data events. You can select multiple buckets in this window. Choose **Add bucket** to log data events for more buckets. Choose to log **Read** events, such as `GetObject`, **Write** events, such as `PutObject`, or both.
    e. Choose **Update trail**.

## WKLD.08 Encrypt Amazon EBS volumes

Enforce encryption of Amazon Elastic Block Store (Amazon EBS) volumes as the default behavior in your AWS account. Encrypted volumes have the same input/output operations per second (IOPS) performance as unencrypted volumes with a minimal effect on latency. This prevents rebuilding volumes at a later date for compliance or other reasons. For more information, see Must-know best practices for Amazon EBS encryption (AWS blog post).

**To encrypt Amazon EBS volumes**

- Enable encryption by default (Amazon EBS documentation).

## WKLD.09 Encrypt Amazon RDS databases

Similar to WKLD.08 Encrypt Amazon EBS volumes, enable encryption of Amazon Relational Database Service (Amazon RDS) databases. This encryption is performed at the underlying volume level and has the same IOPS performance as unencrypted volumes with a minimal effect on latency. For more information, see Overview of encrypting Amazon RDS resources (Amazon RDS documentation).

**To encrypt an RDS database instance**

- Encrypt a database instance (Amazon RDS documentation).

## WKLD.10 Deploy private resources into private subnets

Deploy resources that don't require direct internet access, such as EC2 instances, databases, queues, caching, or other infrastructure, into a VPC private subnet. Private subnets don't have a route declared in their route table to an attached internet gateway and cannot receive internet traffic. Traffic originating from a private subnet that is destined for the internet must undergo network address translation (NAT) through either a managed AWS NAT Gateway or an EC2 instance running NAT processes in a public subnet. For more information about network isolation, see Infrastructure security in Amazon VPC (Amazon VPC documentation).

Use the following practices when creating private resources and subnets:

- When creating a private subnet, disable **auto-assign public IPv4 address**.
- When creating private EC2 instances, disable **Auto-assign Public IP**. This prevents a public IP from being assigned if the instance is unintentionally deployed into a public subnet via misconfiguration.

You specify the subnet for a resource as part of its configuration, when required.

## WKLD.11 Restrict network access by using security groups

Use security groups to control traffic to EC2 instances, RDS databases, and other supported resources. Security groups act as a virtual firewall that can be applied to any group of related resources in order to consistently define rules for allowing inbound and outbound traffic. In addition to rules based on IP addresses and ports, security groups support rules to allow traffic from resources associated to other security groups. For example, a database security group can have rules to allow only traffic from an application server security group.

By default, security groups allow all outbound traffic but don't allow inbound traffic. The outbound traffic rule can be removed, or you can configure additional rules added to restrict outbound traffic and allow inbound traffic. If the security group has no outbound rules, no outbound traffic originating from your instance is allowed. For more information, see Control traffic to resources using security groups (Amazon VPC documentation).

In the following example, there are three security groups that control traffic from an Application Load Balancer to EC2 instances that connect to an Amazon RDS for MySQL database.

| Security group | Inbound rules | Outbound rules |
| --- | --- | --- |
| Application Load Balancer security group | **Description:** Allow HTTPS traffic from anywhere<br>**Type:** HTTPS<br>**Source:** Anywhere-IPv4 (0.0.0.0/0) | **Description:** Allow all traffic to anywhere<br>**Type:** All traffic<br>**Destination:** Anywhere-IPv4 (0.0.0.0/0) |
| EC2 instance security group | **Description:** Allow HTTP traffic from the Application Load Balancer<br>**Type:** HTTP<br>**Source:** Application Load Balancer security group | **Description:** Allow all traffic to anywhere<br>**Type:** All traffic<br>**Destination:** Anywhere-IPv4 (0.0.0.0/0) |
| RDS database security group | **Description:** Allow MySQL traffic from EC2 instance<br>**Type:** MySQL<br>**Source:** EC2 instance security group | No outbound rules |

# WKLD.12 Use VPC endpoints to access supported services

In VPCs, resources that need to access AWS or other external services require either a route to the internet (`0.0.0.0/0`) or to the public IP address of the target service. Use VPC endpoints to enable a private IP route from your VPC to supported AWS or other services, preventing the need to use an internet gateway, NAT device, virtual private network (VPN) connection, or AWS Direct Connect connection.

VPC endpoints support attaching policies and security groups to further control access to a service. For example, you can write a VPC endpoint policy for Amazon DynamoDB to allow only item-level actions and prevent table-level actions for all resources in the VPC, regardless of their own permission policy. You can also write an S3 bucket policy to allow only requests originating from a specific VPC endpoint, denying all other external access. A VPC endpoint can also have a security group rule that, for example, restricts access to only EC2 instances that are associated to an application-specific security group, such as the business-logic tier of a web application.

There are different kinds of VPC endpoints. You access most services by using a VPC interface endpoint. DynamoDB is accessed using a gateway endpoint. Amazon S3 supports both interface and gateway endpoints. Gateway endpoints are recommended for workloads contained within a single AWS account and Region, and come at no additional charge. Interface endpoints are recommended if more extensible access is required, such as to an S3 bucket from other VPCs, from on-premises networks, or from different AWS Regions. Interface endpoints incur an hourly uptime charge and a per-GB data-processing charge, both of which are lower than the respective charges for sending the data to `0.0.0.0/0` through AWS NAT Gateway.

See the following resources for additional information about using VPC endpoints:

- For more information about selecting between gateway and interface endpoints for Amazon S3, see Choosing Your VPC Endpoint Strategy for Amazon S3 (AWS blog post).
- Access an AWS service using an interface VPC endpoint (Amazon VPC documentation).
- Gateway endpoints (Amazon VPC documentation).
- For example S3 bucket policies that restrict access to a specific VPC or VPC endpoint, see Restricting access to a specific VPC (Amazon S3 documentation).
- For example DynamoDB endpoint policies that restrict actions, see Endpoint policies for DynamoDB (Amazon VPC documentation).

## WKLD.13 Require HTTPS for all public web endpoints

Require HTTPS to provide additional credibility to your web endpoints, allow your endpoints to use certificates to prove their identity, and confirm that all traffic between your endpoint and connected clients is encrypted. For public websites, this provides the additional benefit of higher search engine ranking.

Many AWS services provide public web endpoints for your resources, such as AWS Elastic Beanstalk, Amazon CloudFront, Amazon API Gateway, Elastic Load Balancing, and AWS Amplify. For instructions about how require HTTPS for each of these services, see the following:

- Elastic Beanstalk (Elastic Beanstalk documentation)
- CloudFront (CloudFront documentation)
- Application Load Balancer (AWS Knowledge Center)
- Classic Load Balancer (AWS Knowledge Center)
- Amplify (Amplify documentation)

Static websites hosted on Amazon S3 do not support HTTPS. To require HTTPS for these websites, you can use CloudFront. Public access to S3 buckets that are serving content through CloudFront is not required.

**To use CloudFront to serve a static website hosted on Amazon S3**

1. Use CloudFront to serve a static website hosted on Amazon S3 (AWS Knowledge Center).
2. If you are configuring access to a public S3 bucket, require HTTPS between viewers and CloudFront (CloudFront documentation).

   If you are configuring access to a private S3 bucket, restrict access to Amazon S3 content by using an origin access identity (CloudFront documentation).

In addition, configure HTTPS endpoints to require modern Transport Layer Security (TLS) protocols and ciphers, unless compatibility with older protocols is needed. For example, use the `ELBSecurityPolicy-FS-1-2-Res-2020-10` or the most recent policy available for Application Load Balancer HTTPS listeners, instead of the default `ELBSecurityPolicy-2016-08`. The most current policies require TLS 1.2 at minimum, forward secrecy, and strong ciphers that are compatible with modern web browsers.

For more information about the available security policies for HTTPS public endpoints, see:

- Predefined SSL security policies for Classic Load Balancers (Elastic Load Balancing documentation)
- Security policies for your Application Load Balancer (Elastic Load Balancing documentation)
- Supported protocols and ciphers between viewers and CloudFront (CloudFront documentation)

# WKLD.14 Use edge-protection services for public endpoints

Rather than serve traffic direct from compute services such as EC2 instances or containers, use an edge-protection service. This provides an additional layer of security between incoming traffic from the internet and your resources that serve that traffic. These services can filter unwanted traffic, enforce encryption, and apply routing or other rules, such as load balancing, before traffic reaches your internal resources.

AWS services that can provide public endpoint protection include the AWS WAF, CloudFront, Elastic Load Balancing, API Gateway, and Amplify Hosting. Run VPC-based services, such as Elastic Load Balancing, in a public subnet as a proxy to web service resources running in a private subnet.

CloudFront, API Gateway, and Amazon Route 53 provide protection from Layer 3 and 4 distributed denial of service (DDoS) attacks at no charge, and AWS WAF can protect against Layer 7 attacks.

Instructions for getting started with each of these services can be found here:

- Getting Started with AWS WAF (AWS website)
- Getting started with Amazon CloudFront (CloudFront documentation)
- Getting started with Elastic Load Balancing (Elastic Load Balancing documentation)
- Getting started with API Gateway (API Gateway documentation)
- Getting started with Amplify Hosting (Amplify documentation)

# WKLD.15 Define security controls in templates and deploy them by using CI/CD practices

*Infrastructure as code* (IaC) is the practice of defining all of your AWS service resources and configurations in templates and code that you deploy by using continuous integration and continuous delivery (CI/CD) pipelines, the same pipelines used to deploy software applications. IaC services, such as AWS CloudFormation, support both IAM identity-based and resource-based policies and support AWS security services, such as Amazon GuardDuty, AWS WAF, and Amazon VPC. Capture these artifacts as IaC templates, commit the templates to a source code repository, and then deploy them by using CI/CD pipelines.

Unless required otherwise, commit application permission policies with application code in the same repository, and manage general resource policies and security service configurations in separate code repositories and deployment pipelines.

For more information about getting started with IaC on AWS, see the AWS Cloud Development Kit (AWS CDK) documentation.

# Contributors

Contributors to this document include:

- Jay Michael, Principal Solutions Architect (principal author)
- Cole Calistra, Principal Solutions Architect
- Justin Plock, Principal Solutions Architect
- Faisal Farooq, Solutions Architect
- Michael Nguyen, Sr. Solutions Architect
- Ritik Khatwani, Sr. Solutions Architect
- Paul Hawkins, Principal, Office of the Chief Information Security Officer (CISO)

A special thank you to the following people who also helped with guidance and review:

- Robert Put
- Mike Sullivan
- Bob Lee III

# Document history

The following table describes significant changes to this guide. If you want to be notified about future updates, you can subscribe to an RSS feed.

| Change | Description | Date |
| --- | --- | --- |
| Amazon S3 bucket settings | We updated the ACCT.08 Prevent public access to private S3 buckets section to reflect that Amazon S3 buckets created after April 28, 2023 have the **Block Public Access** setting enabled by default. | May 18, 2023 |
| IAM security best practices | We updated this guide for alignment with the latest AWS Identity and Access Management (IAM) best practices. For more informati on, see Security best practices in the IAM documentation. | February 1, 2023 |
| IAM roles | We provided additional links to AWS service documenta tion in the WKLD.01 Use IAM roles for compute environme nt permissions section. | September 22, 2022 |
| Password policy | We updated the recommend ations for strong passwords to use the latest guidance from the Center for Internet Security (CIS). | May 10, 2022 |
| Initial publication | — | April 13, 2022 |

# AWS Prescriptive Guidance glossary

The following are commonly used terms in strategies, guides, and patterns provided by AWS Prescriptive Guidance. To suggest entries, please use the **Provide feedback** link at the end of the glossary.

## Numbers

7 Rs

Seven common migration strategies for moving applications to the cloud. These strategies build upon the 5 Rs that Gartner identified in 2011 and consist of the following:

- Refactor/re-architect – Move an application and modify its architecture by taking full advantage of cloud-native features to improve agility, performance, and scalability. This typically involves porting the operating system and database. Example: Migrate your on-premises Oracle database to the Amazon Aurora PostgreSQL-Compatible Edition.
- Replatform (lift and reshape) – Move an application to the cloud, and introduce some level of optimization to take advantage of cloud capabilities. Example: Migrate your on-premises Oracle database to Amazon Relational Database Service (Amazon RDS) for Oracle in the AWS Cloud.
- Repurchase (drop and shop) – Switch to a different product, typically by moving from a traditional license to a SaaS model. Example: Migrate your customer relationship management (CRM) system to Salesforce.com.
- Rehost (lift and shift) – Move an application to the cloud without making any changes to take advantage of cloud capabilities. Example: Migrate your on-premises Oracle database to Oracle on an EC2 instance in the AWS Cloud.
- Relocate (hypervisor-level lift and shift) – Move infrastructure to the cloud without purchasing new hardware, rewriting applications, or modifying your existing operations. You migrate servers from an on-premises platform to a cloud service for the same platform. Example: Migrate a Microsoft Hyper-V application to AWS.
- Retain (revisit) – Keep applications in your source environment. These might include applications that require major refactoring, and you want to postpone that work until a later time, and legacy applications that you want to retain, because there's no business justification for migrating them.

- Retire – Decommission or remove applications that are no longer needed in your source environment.

# A

ABAC

See attribute-based access control.

abstracted services

See managed services.

ACID

See atomicity, consistency, isolation, durability.

active-active migration

A database migration method in which the source and target databases are kept in sync (by using a bidirectional replication tool or dual write operations), and both databases handle transactions from connecting applications during migration. This method supports migration in small, controlled batches instead of requiring a one-time cutover. It's more flexible but requires more work than active-passive migration.

active-passive migration

A database migration method in which in which the source and target databases are kept in sync, but only the source database handles transactions from connecting applications while data is replicated to the target database. The target database doesn't accept any transactions during migration.

aggregate function

A SQL function that operates on a group of rows and calculates a single return value for the group. Examples of aggregate functions include `SUM` and `MAX`.

AI

See artificial intelligence.

AIOps

See artificial intelligence operations.

anonymization

The process of permanently deleting personal information in a dataset. Anonymization can help protect personal privacy. Anonymized data is no longer considered to be personal data.

anti-pattern

A frequently used solution for a recurring issue where the solution is counter-productive, ineffective, or less effective than an alternative.

application control

A security approach that allows the use of only approved applications in order to help protect a system from malware.

application portfolio

A collection of detailed information about each application used by an organization, including the cost to build and maintain the application, and its business value. This information is key to the portfolio discovery and analysis process and helps identify and prioritize the applications to be migrated, modernized, and optimized.

artificial intelligence (AI)

The field of computer science that is dedicated to using computing technologies to perform cognitive functions that are typically associated with humans, such as learning, solving problems, and recognizing patterns. For more information, see What is Artificial Intelligence?

artificial intelligence operations (AIOps)

The process of using machine learning techniques to solve operational problems, reduce operational incidents and human intervention, and increase service quality. For more information about how AIOps is used in the AWS migration strategy, see the operations integration guide.

asymmetric encryption

An encryption algorithm that uses a pair of keys, a public key for encryption and a private key for decryption. You can share the public key because it isn't used for decryption, but access to the private key should be highly restricted.

atomicity, consistency, isolation, durability (ACID)

A set of software properties that guarantee the data validity and operational reliability of a database, even in the case of errors, power failures, or other problems.

attribute-based access control (ABAC)

The practice of creating fine-grained permissions based on user attributes, such as department, job role, and team name. For more information, see ABAC for AWS in the AWS Identity and Access Management (IAM) documentation.

authoritative data source

A location where you store the primary version of data, which is considered to be the most reliable source of information. You can copy data from the authoritative data source to other locations for the purposes of processing or modifying the data, such as anonymizing, redacting, or pseudonymizing it.

Availability Zone

A distinct location within an AWS Region that is insulated from failures in other Availability Zones and provides inexpensive, low-latency network connectivity to other Availability Zones in the same Region.

AWS Cloud Adoption Framework (AWS CAF)

A framework of guidelines and best practices from AWS to help organizations develop an efficient and effective plan to move successfully to the cloud. AWS CAF organizes guidance into six focus areas called perspectives: business, people, governance, platform, security, and operations. The business, people, and governance perspectives focus on business skills and processes; the platform, security, and operations perspectives focus on technical skills and processes. For example, the people perspective targets stakeholders who handle human resources (HR), staffing functions, and people management. For this perspective, AWS CAF provides guidance for people development, training, and communications to help ready the organization for successful cloud adoption. For more information, see the AWS CAF website and the AWS CAF whitepaper.

AWS Workload Qualification Framework (AWS WQF)

A tool that evaluates database migration workloads, recommends migration strategies, and provides work estimates. AWS WQF is included with AWS Schema Conversion Tool (AWS SCT). It analyzes database schemas and code objects, application code, dependencies, and performance characteristics, and provides assessment reports.

# B

bad bot

A bot that is intended to disrupt or cause harm to individuals or organizations.

BCP

See business continuity planning.

behavior graph

A unified, interactive view of resource behavior and interactions over time. You can use a behavior graph with Amazon Detective to examine failed logon attempts, suspicious API calls, and similar actions. For more information, see Data in a behavior graph in the Detective documentation.

big-endian system

A system that stores the most significant byte first. See also endianness.

binary classification

A process that predicts a binary outcome (one of two possible classes). For example, your ML model might need to predict problems such as “Is this email spam or not spam?" or "Is this product a book or a car?"

bloom filter

A probabilistic, memory-efficient data structure that is used to test whether an element is a member of a set.

blue/green deployment

A deployment strategy where you create two separate but identical environments. You run the current application version in one environment (blue) and the new application version in the other environment (green). This strategy helps you quickly roll back with minimal impact.

bot

A software application that runs automated tasks over the internet and simulates human activity or interaction. Some bots are useful or beneficial, such as web crawlers that index information on the internet. Some other bots, known as *bad bots*, are intended to disrupt or cause harm to individuals or organizations.

botnet

Networks of bots that are infected by malware and are under the control of a single party, known as a *bot herder* or *bot operator*. Botnets are the best-known mechanism to scale bots and their impact.

branch

A contained area of a code repository. The first branch created in a repository is the *main branch*. You can create a new branch from an existing branch, and you can then develop features or fix bugs in the new branch. A branch you create to build a feature is commonly referred to as a *feature branch*. When the feature is ready for release, you merge the feature branch back into the main branch. For more information, see About branches (GitHub documentation).

break-glass access

In exceptional circumstances and through an approved process, a quick means for a user to gain access to an AWS account that they don't typically have permissions to access. For more information, see the Implement break-glass procedures indicator in the AWS Well-Architected guidance.

brownfield strategy

The existing infrastructure in your environment. When adopting a brownfield strategy for a system architecture, you design the architecture around the constraints of the current systems and infrastructure. If you are expanding the existing infrastructure, you might blend brownfield and greenfield strategies.

buffer cache

The memory area where the most frequently accessed data is stored.

business capability

What a business does to generate value (for example, sales, customer service, or marketing). Microservices architectures and development decisions can be driven by business capabilities. For more information, see the Organized around business capabilities section of the Running containerized microservices on AWS whitepaper.

business continuity planning (BCP)

A plan that addresses the potential impact of a disruptive event, such as a large-scale migration, on operations and enables a business to resume operations quickly.

# C

CAF

See AWS Cloud Adoption Framework.

canary deployment

The slow and incremental release of a version to end users. When you are confident, you deploy the new version and replace the current version in its entirety.

CCoE

See Cloud Center of Excellence.

CDC

See change data capture.

change data capture (CDC)

The process of tracking changes to a data source, such as a database table, and recording metadata about the change. You can use CDC for various purposes, such as auditing or replicating changes in a target system to maintain synchronization.

chaos engineering

Intentionally introducing failures or disruptive events to test a system's resilience. You can use AWS Fault Injection Service (AWS FIS) to perform experiments that stress your AWS workloads and evaluate their response.

CI/CD

See continuous integration and continuous delivery.

classification

A categorization process that helps generate predictions. ML models for classification problems predict a discrete value. Discrete values are always distinct from one another. For example, a model might need to evaluate whether or not there is a car in an image.

client-side encryption

Encryption of data locally, before the target AWS service receives it.

Cloud Center of Excellence (CCoE)

A multi-disciplinary team that drives cloud adoption efforts across an organization, including developing cloud best practices, mobilizing resources, establishing migration timelines, and leading the organization through large-scale transformations. For more information, see the CCoE posts on the AWS Cloud Enterprise Strategy Blog.

cloud computing

The cloud technology that is typically used for remote data storage and IoT device management. Cloud computing is commonly connected to edge computing technology.

cloud operating model

In an IT organization, the operating model that is used to build, mature, and optimize one or more cloud environments. For more information, see Building your Cloud Operating Model.

cloud stages of adoption

The four phases that organizations typically go through when they migrate to the AWS Cloud:

- Project – Running a few cloud-related projects for proof of concept and learning purposes
- Foundation – Making foundational investments to scale your cloud adoption (e.g., creating a landing zone, defining a CCoE, establishing an operations model)
- Migration – Migrating individual applications
- Re-invention – Optimizing products and services, and innovating in the cloud

These stages were defined by Stephen Orban in the blog post The Journey Toward Cloud-First & the Stages of Adoption on the AWS Cloud Enterprise Strategy blog. For information about how they relate to the AWS migration strategy, see the migration readiness guide.

CMDB

See configuration management database.

code repository

A location where source code and other assets, such as documentation, samples, and scripts, are stored and updated through version control processes. Common cloud repositories include GitHub or Bitbucket Cloud. Each version of the code is called a *branch*. In a microservice structure, each repository is devoted to a single piece of functionality. A single CI/CD pipeline can use multiple repositories.

cold cache

A buffer cache that is empty, not well populated, or contains stale or irrelevant data. This affects performance because the database instance must read from the main memory or disk, which is slower than reading from the buffer cache.

cold data

Data that is rarely accessed and is typically historical. When querying this kind of data, slow queries are typically acceptable. Moving this data to lower-performing and less expensive storage tiers or classes can reduce costs.

computer vision (CV)

A field of AI that uses machine learning to analyze and extract information from visual formats such as digital images and videos. For example, Amazon SageMaker AI provides image processing algorithms for CV.

configuration drift

For a workload, a configuration change from the expected state. It might cause the workload to become noncompliant, and it's typically gradual and unintentional.

configuration management database (CMDB)

A repository that stores and manages information about a database and its IT environment, including both hardware and software components and their configurations. You typically use data from a CMDB in the portfolio discovery and analysis stage of migration.

conformance pack

A collection of AWS Config rules and remediation actions that you can assemble to customize your compliance and security checks. You can deploy a conformance pack as a single entity in an AWS account and Region, or across an organization, by using a YAML template. For more information, see Conformance packs in the AWS Config documentation.

continuous integration and continuous delivery (CI/CD)

The process of automating the source, build, test, staging, and production stages of the software release process. CI/CD is commonly described as a pipeline. CI/CD can help you automate processes, improve productivity, improve code quality, and deliver faster. For more information, see Benefits of continuous delivery. CD can also stand for *continuous deployment*. For more information, see Continuous Delivery vs. Continuous Deployment.

CV

See computer vision.

# D

data at rest

Data that is stationary in your network, such as data that is in storage.

data classification

A process for identifying and categorizing the data in your network based on its criticality and sensitivity. It is a critical component of any cybersecurity risk management strategy because it helps you determine the appropriate protection and retention controls for the data. Data classification is a component of the security pillar in the AWS Well-Architected Framework. For more information, see Data classification.

data drift

A meaningful variation between the production data and the data that was used to train an ML model, or a meaningful change in the input data over time. Data drift can reduce the overall quality, accuracy, and fairness in ML model predictions.

data in transit

Data that is actively moving through your network, such as between network resources.

data mesh

An architectural framework that provides distributed, decentralized data ownership with centralized management and governance.

data minimization

The principle of collecting and processing only the data that is strictly necessary. Practicing data minimization in the AWS Cloud can reduce privacy risks, costs, and your analytics carbon footprint.

data perimeter

A set of preventive guardrails in your AWS environment that help make sure that only trusted identities are accessing trusted resources from expected networks. For more information, see Building a data perimeter on AWS.

data preprocessing

To transform raw data into a format that is easily parsed by your ML model. Preprocessing data can mean removing certain columns or rows and addressing missing, inconsistent, or duplicate values.

data provenance

The process of tracking the origin and history of data throughout its lifecycle, such as how the data was generated, transmitted, and stored.

data subject

An individual whose data is being collected and processed.

data warehouse

A data management system that supports business intelligence, such as analytics. Data warehouses commonly contain large amounts of historical data, and they are typically used for queries and analysis.

database definition language (DDL)

Statements or commands for creating or modifying the structure of tables and objects in a database.

database manipulation language (DML)

Statements or commands for modifying (inserting, updating, and deleting) information in a database.

DDL

See database definition language.

deep ensemble

To combine multiple deep learning models for prediction. You can use deep ensembles to obtain a more accurate prediction or for estimating uncertainty in predictions.

deep learning

An ML subfield that uses multiple layers of artificial neural networks to identify mapping between input data and target variables of interest.

defense-in-depth

An information security approach in which a series of security mechanisms and controls are thoughtfully layered throughout a computer network to protect the confidentiality, integrity, and availability of the network and the data within. When you adopt this strategy on AWS, you add multiple controls at different layers of the AWS Organizations structure to help secure resources. For example, a defense-in-depth approach might combine multi-factor authentication, network segmentation, and encryption.

delegated administrator

In AWS Organizations, a compatible service can register an AWS member account to administer the organization's accounts and manage permissions for that service. This account is called the *delegated administrator* for that service. For more information and a list of compatible services, see Services that work with AWS Organizations in the AWS Organizations documentation.

deployment

The process of making an application, new features, or code fixes available in the target environment. Deployment involves implementing changes in a code base and then building and running that code base in the application's environments.

development environment

See environment.

detective control

A security control that is designed to detect, log, and alert after an event has occurred. These controls are a second line of defense, alerting you to security events that bypassed the preventative controls in place. For more information, see Detective controls in *Implementing security controls on AWS*.

development value stream mapping (DVSM)

A process used to identify and prioritize constraints that adversely affect speed and quality in a software development lifecycle. DVSM extends the value stream mapping process originally designed for lean manufacturing practices. It focuses on the steps and teams required to create and move value through the software development process.

digital twin

A virtual representation of a real-world system, such as a building, factory, industrial equipment, or production line. Digital twins support predictive maintenance, remote monitoring, and production optimization.

dimension table

In a star schema, a smaller table that contains data attributes about quantitative data in a fact table. Dimension table attributes are typically text fields or discrete numbers that behave like text. These attributes are commonly used for query constraining, filtering, and result set labeling.

disaster

An event that prevents a workload or system from fulfilling its business objectives in its primary deployed location. These events can be natural disasters, technical failures, or the result of human actions, such as unintentional misconfiguration or a malware attack.

disaster recovery (DR)

The strategy and process you use to minimize downtime and data loss caused by a disaster. For more information, see Disaster Recovery of Workloads on AWS: Recovery in the Cloud in the AWS Well-Architected Framework.

DML

See database manipulation language.

domain-driven design

An approach to developing a complex software system by connecting its components to evolving domains, or core business goals, that each component serves. This concept was introduced by Eric Evans in his book, *Domain-Driven Design: Tackling Complexity in the Heart of Software* (Boston: Addison-Wesley Professional, 2003). For information about how you can use domain-driven design with the strangler fig pattern, see Modernizing legacy Microsoft ASP.NET (ASMX) web services incrementally by using containers and Amazon API Gateway.

DR

See disaster recovery.

drift detection

Tracking deviations from a baselined configuration. For example, you can use AWS CloudFormation to detect drift in system resources, or you can use AWS Control Tower to detect changes in your landing zone that might affect compliance with governance requirements.

DVSM

See development value stream mapping.

# E

EDA

See exploratory data analysis.

EDI

See electronic data interchange.

edge computing

The technology that increases the computing power for smart devices at the edges of an IoT network. When compared with cloud computing, edge computing can reduce communication latency and improve response time.

electronic data interchange (EDI)

The automated exchange of business documents between organizations. For more information, see What is Electronic Data Interchange.

encryption

A computing process that transforms plaintext data, which is human-readable, into ciphertext.

encryption key

A cryptographic string of randomized bits that is generated by an encryption algorithm. Keys can vary in length, and each key is designed to be unpredictable and unique.

endianness

The order in which bytes are stored in computer memory. Big-endian systems store the most significant byte first. Little-endian systems store the least significant byte first.

endpoint

See service endpoint.

endpoint service

A service that you can host in a virtual private cloud (VPC) to share with other users. You can create an endpoint service with AWS PrivateLink and grant permissions to other AWS accounts or to AWS Identity and Access Management (IAM) principals. These accounts or principals can connect to your endpoint service privately by creating interface VPC endpoints. For more

information, see Create an endpoint service in the Amazon Virtual Private Cloud (Amazon VPC) documentation.

enterprise resource planning (ERP)

A system that automates and manages key business processes (such as accounting, MES, and project management) for an enterprise.

envelope encryption

The process of encrypting an encryption key with another encryption key. For more information, see Envelope encryption in the AWS Key Management Service (AWS KMS) documentation.

environment

An instance of a running application. The following are common types of environments in cloud computing:

- development environment – An instance of a running application that is available only to the core team responsible for maintaining the application. Development environments are used to test changes before promoting them to upper environments. This type of environment is sometimes referred to as a *test environment*.
- lower environments – All development environments for an application, such as those used for initial builds and tests.
- production environment – An instance of a running application that end users can access. In a CI/CD pipeline, the production environment is the last deployment environment.
- upper environments – All environments that can be accessed by users other than the core development team. This can include a production environment, preproduction environments, and environments for user acceptance testing.

epic

In agile methodologies, functional categories that help organize and prioritize your work. Epics provide a high-level description of requirements and implementation tasks. For example, AWS CAF security epics include identity and access management, detective controls, infrastructure security, data protection, and incident response. For more information about epics in the AWS migration strategy, see the program implementation guide.

ERP

See enterprise resource planning.

exploratory data analysis (EDA)

The process of analyzing a dataset to understand its main characteristics. You collect or aggregate data and then perform initial investigations to find patterns, detect anomalies, and check assumptions. EDA is performed by calculating summary statistics and creating data visualizations.

# F

fact table

The central table in a star schema. It stores quantitative data about business operations. Typically, a fact table contains two types of columns: those that contain measures and those that contain a foreign key to a dimension table.

fail fast

A philosophy that uses frequent and incremental testing to reduce the development lifecycle. It is a critical part of an agile approach.

fault isolation boundary

In the AWS Cloud, a boundary such as an Availability Zone, AWS Region, control plane, or data plane that limits the effect of a failure and helps improve the resilience of workloads. For more information, see AWS Fault Isolation Boundaries.

feature branch

See branch.

features

The input data that you use to make a prediction. For example, in a manufacturing context, features could be images that are periodically captured from the manufacturing line.

feature importance

How significant a feature is for a model's predictions. This is usually expressed as a numerical score that can be calculated through various techniques, such as Shapley Additive Explanations (SHAP) and integrated gradients. For more information, see Machine learning model interpretability with AWS.

feature transformation

To optimize data for the ML process, including enriching data with additional sources, scaling values, or extracting multiple sets of information from a single data field. This enables the ML model to benefit from the data. For example, if you break down the “2021-05-27 00:15:37” date into “2021”, “May”, “Thu”, and “15”, you can help the learning algorithm learn nuanced patterns associated with different data components.

few-shot prompting

Providing an LLM with a small number of examples that demonstrate the task and desired output before asking it to perform a similar task. This technique is an application of in-context learning, where models learn from examples (*shots*) that are embedded in prompts. Few-shot prompting can be effective for tasks that require specific formatting, reasoning, or domain knowledge. See also zero-shot prompting.

FGAC

See fine-grained access control.

fine-grained access control (FGAC)

The use of multiple conditions to allow or deny an access request.

flash-cut migration

A database migration method that uses continuous data replication through change data capture to migrate data in the shortest time possible, instead of using a phased approach. The objective is to keep downtime to a minimum.

FM

See foundation model.

foundation model (FM)

A large deep-learning neural network that has been training on massive datasets of generalized and unlabeled data. FMs are capable of performing a wide variety of general tasks, such as understanding language, generating text and images, and conversing in natural language. For more information, see What are Foundation Models.

# G

generative AI

A subset of AI models that have been trained on large amounts of data and that can use a simple text prompt to create new content and artifacts, such as images, videos, text, and audio. For more information, see What is Generative AI.

geo blocking

See geographic restrictions.

geographic restrictions (geo blocking)

In Amazon CloudFront, an option to prevent users in specific countries from accessing content distributions. You can use an allow list or block list to specify approved and banned countries. For more information, see Restricting the geographic distribution of your content in the CloudFront documentation.

Gitflow workflow

An approach in which lower and upper environments use different branches in a source code repository. The Gitflow workflow is considered legacy, and the trunk-based workflow is the modern, preferred approach.

golden image

A snapshot of a system or software that is used as a template to deploy new instances of that system or software. For example, in manufacturing, a golden image can be used to provision software on multiple devices and helps improve speed, scalability, and productivity in device manufacturing operations.

greenfield strategy

The absence of existing infrastructure in a new environment. When adopting a greenfield strategy for a system architecture, you can select all new technologies without the restriction of compatibility with existing infrastructure, also known as brownfield. If you are expanding the existing infrastructure, you might blend brownfield and greenfield strategies.

guardrail

A high-level rule that helps govern resources, policies, and compliance across organizational units (OUs). *Preventive guardrails* enforce policies to ensure alignment to compliance standards. They are implemented by using service control policies and IAM permissions boundaries.

*Detective guardrails* detect policy violations and compliance issues, and generate alerts for remediation. They are implemented by using AWS Config, AWS Security Hub, Amazon GuardDuty, AWS Trusted Advisor, Amazon Inspector, and custom AWS Lambda checks.

# H

HA

See high availability.

heterogeneous database migration

Migrating your source database to a target database that uses a different database engine (for example, Oracle to Amazon Aurora). Heterogeneous migration is typically part of a re-architecting effort, and converting the schema can be a complex task. AWS provides AWS SCT that helps with schema conversions.

high availability (HA)

The ability of a workload to operate continuously, without intervention, in the event of challenges or disasters. HA systems are designed to automatically fail over, consistently deliver high-quality performance, and handle different loads and failures with minimal performance impact.

historian modernization

An approach used to modernize and upgrade operational technology (OT) systems to better serve the needs of the manufacturing industry. A *historian* is a type of database that is used to collect and store data from various sources in a factory.

holdout data

A portion of historical, labeled data that is withheld from a dataset that is used to train a machine learning model. You can use holdout data to evaluate the model performance by comparing the model predictions against the holdout data.

homogeneous database migration

Migrating your source database to a target database that shares the same database engine (for example, Microsoft SQL Server to Amazon RDS for SQL Server). Homogeneous migration is typically part of a rehosting or replatforming effort. You can use native database utilities to migrate the schema.

hot data

Data that is frequently accessed, such as real-time data or recent translational data. This data typically requires a high-performance storage tier or class to provide fast query responses.

hotfix

An urgent fix for a critical issue in a production environment. Due to its urgency, a hotfix is usually made outside of the typical DevOps release workflow.

hypercare period

Immediately following cutover, the period of time when a migration team manages and monitors the migrated applications in the cloud in order to address any issues. Typically, this period is 1–4 days in length. At the end of the hypercare period, the migration team typically transfers responsibility for the applications to the cloud operations team.

## I

IaC

See infrastructure as code.

identity-based policy

A policy attached to one or more IAM principals that defines their permissions within the AWS Cloud environment.

idle application

An application that has an average CPU and memory usage between 5 and 20 percent over a period of 90 days. In a migration project, it is common to retire these applications or retain them on premises.

IIoT

See industrial Internet of Things.

immutable infrastructure

A model that deploys new infrastructure for production workloads instead of updating, patching, or modifying the existing infrastructure. Immutable infrastructures are inherently more consistent, reliable, and predictable than mutable infrastructure. For more information, see the Deploy using immutable infrastructure best practice in the AWS Well-Architected Framework.

inbound (ingress) VPC

In an AWS multi-account architecture, a VPC that accepts, inspects, and routes network connections from outside an application. The AWS Security Reference Architecture recommends setting up your Network account with inbound, outbound, and inspection VPCs to protect the two-way interface between your application and the broader internet.

incremental migration

A cutover strategy in which you migrate your application in small parts instead of performing a single, full cutover. For example, you might move only a few microservices or users to the new system initially. After you verify that everything is working properly, you can incrementally move additional microservices or users until you can decommission your legacy system. This strategy reduces the risks associated with large migrations.

Industry 4.0

A term that was introduced by Klaus Schwab in 2016 to refer to the modernization of manufacturing processes through advances in connectivity, real-time data, automation, analytics, and AI/ML.

infrastructure

All of the resources and assets contained within an application's environment.

infrastructure as code (IaC)

The process of provisioning and managing an application's infrastructure through a set of configuration files. IaC is designed to help you centralize infrastructure management, standardize resources, and scale quickly so that new environments are repeatable, reliable, and consistent.

industrial Internet of Things (IIoT)

The use of internet-connected sensors and devices in the industrial sectors, such as manufacturing, energy, automotive, healthcare, life sciences, and agriculture. For more information, see Building an industrial Internet of Things (IIoT) digital transformation strategy.

inspection VPC

In an AWS multi-account architecture, a centralized VPC that manages inspections of network traffic between VPCs (in the same or different AWS Regions), the internet, and on-premises networks. The AWS Security Reference Architecture recommends setting up your Network account with inbound, outbound, and inspection VPCs to protect the two-way interface between your application and the broader internet.

Internet of Things (IoT)

The network of connected physical objects with embedded sensors or processors that communicate with other devices and systems through the internet or over a local communication network. For more information, see What is IoT?

interpretability

A characteristic of a machine learning model that describes the degree to which a human can understand how the model's predictions depend on its inputs. For more information, see Machine learning model interpretability with AWS.

IoT

See Internet of Things.

IT information library (ITIL)

A set of best practices for delivering IT services and aligning these services with business requirements. ITIL provides the foundation for ITSM.

IT service management (ITSM)

Activities associated with designing, implementing, managing, and supporting IT services for an organization. For information about integrating cloud operations with ITSM tools, see the operations integration guide.

ITIL

See IT information library.

ITSM

See IT service management.

# L

label-based access control (LBAC)

An implementation of mandatory access control (MAC) where the users and the data itself are each explicitly assigned a security label value. The intersection between the user security label and data security label determines which rows and columns can be seen by the user.

landing zone

A landing zone is a well-architected, multi-account AWS environment that is scalable and secure. This is a starting point from which your organizations can quickly launch and deploy workloads and applications with confidence in their security and infrastructure environment. For more information about landing zones, see Setting up a secure and scalable multi-account AWS environment.

large language model (LLM)

A deep learning AI model that is pretrained on a vast amount of data. An LLM can perform multiple tasks, such as answering questions, summarizing documents, translating text into other languages, and completing sentences. For more information, see What are LLMs.

large migration

A migration of 300 or more servers.

LBAC

See label-based access control.

least privilege

The security best practice of granting the minimum permissions required to perform a task. For more information, see Apply least-privilege permissions in the IAM documentation.

lift and shift

See 7 Rs.

little-endian system

A system that stores the least significant byte first. See also endianness.

LLM

See large language model.

lower environments

See environment.

# M

machine learning (ML)

A type of artificial intelligence that uses algorithms and techniques for pattern recognition and learning. ML analyzes and learns from recorded data, such as Internet of Things (IoT) data, to generate a statistical model based on patterns. For more information, see Machine Learning.

main branch

See branch.

malware

Software that is designed to compromise computer security or privacy. Malware might disrupt computer systems, leak sensitive information, or gain unauthorized access. Examples of malware include viruses, worms, ransomware, Trojan horses, spyware, and keyloggers.

managed services

AWS services for which AWS operates the infrastructure layer, the operating system, and platforms, and you access the endpoints to store and retrieve data. Amazon Simple Storage Service (Amazon S3) and Amazon DynamoDB are examples of managed services. These are also known as *abstracted services*.

manufacturing execution system (MES)

A software system for tracking, monitoring, documenting, and controlling production processes that convert raw materials to finished products on the shop floor.

MAP

See Migration Acceleration Program.

mechanism

A complete process in which you create a tool, drive adoption of the tool, and then inspect the results in order to make adjustments. A mechanism is a cycle that reinforces and improves itself as it operates. For more information, see Building mechanisms in the AWS Well-Architected Framework.

member account

All AWS accounts other than the management account that are part of an organization in AWS Organizations. An account can be a member of only one organization at a time.

MES

See manufacturing execution system.

Message Queuing Telemetry Transport (MQTT)

A lightweight, machine-to-machine (M2M) communication protocol, based on the publish/subscribe pattern, for resource-constrained IoT devices.

microservice

A small, independent service that communicates over well-defined APIs and is typically owned by small, self-contained teams. For example, an insurance system might include microservices that map to business capabilities, such as sales or marketing, or subdomains, such as purchasing, claims, or analytics. The benefits of microservices include agility, flexible scaling, easy deployment, reusable code, and resilience. For more information, see Integrating microservices by using AWS serverless services.

microservices architecture

An approach to building an application with independent components that run each application process as a microservice. These microservices communicate through a well-defined interface by using lightweight APIs. Each microservice in this architecture can be updated, deployed, and scaled to meet demand for specific functions of an application. For more information, see Implementing microservices on AWS.

Migration Acceleration Program (MAP)

An AWS program that provides consulting support, training, and services to help organizations build a strong operational foundation for moving to the cloud, and to help offset the initial cost of migrations. MAP includes a migration methodology for executing legacy migrations in a methodical way and a set of tools to automate and accelerate common migration scenarios.

migration at scale

The process of moving the majority of the application portfolio to the cloud in waves, with more applications moved at a faster rate in each wave. This phase uses the best practices and lessons learned from the earlier phases to implement a *migration factory* of teams, tools, and processes to streamline the migration of workloads through automation and agile delivery. This is the third phase of the AWS migration strategy.

migration factory

Cross-functional teams that streamline the migration of workloads through automated, agile approaches. Migration factory teams typically include operations, business analysts and owners,

migration engineers, developers, and DevOps professionals working in sprints. Between 20 and 50 percent of an enterprise application portfolio consists of repeated patterns that can be optimized by a factory approach. For more information, see the discussion of migration factories and the Cloud Migration Factory guide in this content set.

migration metadata

The information about the application and server that is needed to complete the migration. Each migration pattern requires a different set of migration metadata. Examples of migration metadata include the target subnet, security group, and AWS account.

migration pattern

A repeatable migration task that details the migration strategy, the migration destination, and the migration application or service used. Example: Rehost migration to Amazon EC2 with AWS Application Migration Service.

Migration Portfolio Assessment (MPA)

An online tool that provides information for validating the business case for migrating to the AWS Cloud. MPA provides detailed portfolio assessment (server right-sizing, pricing, TCO comparisons, migration cost analysis) as well as migration planning (application data analysis and data collection, application grouping, migration prioritization, and wave planning). The MPA tool (requires login) is available free of charge to all AWS consultants and APN Partner consultants.

Migration Readiness Assessment (MRA)

The process of gaining insights about an organization's cloud readiness status, identifying strengths and weaknesses, and building an action plan to close identified gaps, using the AWS CAF. For more information, see the migration readiness guide. MRA is the first phase of the AWS migration strategy.

migration strategy

The approach used to migrate a workload to the AWS Cloud. For more information, see the 7 Rs entry in this glossary and see Mobilize your organization to accelerate large-scale migrations.

ML

See machine learning.

modernization

Transforming an outdated (legacy or monolithic) application and its infrastructure into an agile, elastic, and highly available system in the cloud to reduce costs, gain efficiencies, and take advantage of innovations. For more information, see Strategy for modernizing applications in the AWS Cloud.

modernization readiness assessment

An evaluation that helps determine the modernization readiness of an organization's applications; identifies benefits, risks, and dependencies; and determines how well the organization can support the future state of those applications. The outcome of the assessment is a blueprint of the target architecture, a roadmap that details development phases and milestones for the modernization process, and an action plan for addressing identified gaps. For more information, see Evaluating modernization readiness for applications in the AWS Cloud.

monolithic applications (monoliths)

Applications that run as a single service with tightly coupled processes. Monolithic applications have several drawbacks. If one application feature experiences a spike in demand, the entire architecture must be scaled. Adding or improving a monolithic application's features also becomes more complex when the code base grows. To address these issues, you can use a microservices architecture. For more information, see Decomposing monoliths into microservices.

MPA

See Migration Portfolio Assessment.

MQTT

See Message Queuing Telemetry Transport.

multiclass classification

A process that helps generate predictions for multiple classes (predicting one of more than two outcomes). For example, an ML model might ask "Is this product a book, car, or phone?" or "Which product category is most interesting to this customer?"

mutable infrastructure

A model that updates and modifies the existing infrastructure for production workloads. For improved consistency, reliability, and predictability, the AWS Well-Architected Framework recommends the use of immutable infrastructure as a best practice.

# O

OAC

See origin access control.

OAI

See origin access identity.

OCM

See organizational change management.

offline migration

A migration method in which the source workload is taken down during the migration process. This method involves extended downtime and is typically used for small, non-critical workloads.

OI

See operations integration.

OLA

See operational-level agreement.

online migration

A migration method in which the source workload is copied to the target system without being taken offline. Applications that are connected to the workload can continue to function during the migration. This method involves zero to minimal downtime and is typically used for critical production workloads.

OPC-UA

See Open Process Communications - Unified Architecture.

Open Process Communications - Unified Architecture (OPC-UA)

A machine-to-machine (M2M) communication protocol for industrial automation. OPC-UA provides an interoperability standard with data encryption, authentication, and authorization schemes.

operational-level agreement (OLA)

An agreement that clarifies what functional IT groups promise to deliver to each other, to support a service-level agreement (SLA).

operational readiness review (ORR)

A checklist of questions and associated best practices that help you understand, evaluate, prevent, or reduce the scope of incidents and possible failures. For more information, see Operational Readiness Reviews (ORR) in the AWS Well-Architected Framework.

operational technology (OT)

Hardware and software systems that work with the physical environment to control industrial operations, equipment, and infrastructure. In manufacturing, the integration of OT and information technology (IT) systems is a key focus for Industry 4.0 transformations.

operations integration (OI)

The process of modernizing operations in the cloud, which involves readiness planning, automation, and integration. For more information, see the operations integration guide.

organization trail

A trail that's created by AWS CloudTrail that logs all events for all AWS accounts in an organization in AWS Organizations. This trail is created in each AWS account that's part of the organization and tracks the activity in each account. For more information, see Creating a trail for an organization in the CloudTrail documentation.

organizational change management (OCM)

A framework for managing major, disruptive business transformations from a people, culture, and leadership perspective. OCM helps organizations prepare for, and transition to, new systems and strategies by accelerating change adoption, addressing transitional issues, and driving cultural and organizational changes. In the AWS migration strategy, this framework is called *people acceleration*, because of the speed of change required in cloud adoption projects. For more information, see the OCM guide.

origin access control (OAC)

In CloudFront, an enhanced option for restricting access to secure your Amazon Simple Storage Service (Amazon S3) content. OAC supports all S3 buckets in all AWS Regions, server-side encryption with AWS KMS (SSE-KMS), and dynamic PUT and DELETE requests to the S3 bucket.

origin access identity (OAI)

In CloudFront, an option for restricting access to secure your Amazon S3 content. When you use OAI, CloudFront creates a principal that Amazon S3 can authenticate with. Authenticated principals can access content in an S3 bucket only through a specific CloudFront distribution. See also OAC, which provides more granular and enhanced access control.

ORR

See operational readiness review.

OT

See operational technology.

outbound (egress) VPC

In an AWS multi-account architecture, a VPC that handles network connections that are initiated from within an application. The AWS Security Reference Architecture recommends setting up your Network account with inbound, outbound, and inspection VPCs to protect the two-way interface between your application and the broader internet.

# P

permissions boundary

An IAM management policy that is attached to IAM principals to set the maximum permissions that the user or role can have. For more information, see Permissions boundaries in the IAM documentation.

personally identifiable information (PII)

Information that, when viewed directly or paired with other related data, can be used to reasonably infer the identity of an individual. Examples of PII include names, addresses, and contact information.

PII

See personally identifiable information.

playbook

A set of predefined steps that capture the work associated with migrations, such as delivering core operations functions in the cloud. A playbook can take the form of scripts, automated runbooks, or a summary of processes or steps required to operate your modernized environment.

PLC

See programmable logic controller.

PLM

See product lifecycle management.

policy

An object that can define permissions (see identity-based policy), specify access conditions (see resource-based policy), or define the maximum permissions for all accounts in an organization in AWS Organizations (see service control policy).

polyglot persistence

Independently choosing a microservice's data storage technology based on data access patterns and other requirements. If your microservices have the same data storage technology, they can encounter implementation challenges or experience poor performance. Microservices are more easily implemented and achieve better performance and scalability if they use the data store best adapted to their requirements. For more information, see Enabling data persistence in microservices.

portfolio assessment

A process of discovering, analyzing, and prioritizing the application portfolio in order to plan the migration. For more information, see Evaluating migration readiness.

predicate

A query condition that returns `true` or `false`, commonly located in a `WHERE` clause.

predicate pushdown

A database query optimization technique that filters the data in the query before transfer. This reduces the amount of data that must be retrieved and processed from the relational database, and it improves query performance.

preventative control

A security control that is designed to prevent an event from occurring. These controls are a first line of defense to help prevent unauthorized access or unwanted changes to your network. For more information, see Preventative controls in *Implementing security controls on AWS*.

principal

An entity in AWS that can perform actions and access resources. This entity is typically a root user for an AWS account, an IAM role, or a user. For more information, see *Principal* in Roles terms and concepts in the IAM documentation.

privacy by design

A system engineering approach that takes privacy into account through the whole development process.

private hosted zones

A container that holds information about how you want Amazon Route 53 to respond to DNS queries for a domain and its subdomains within one or more VPCs. For more information, see Working with private hosted zones in the Route 53 documentation.

proactive control

A security control designed to prevent the deployment of noncompliant resources. These controls scan resources before they are provisioned. If the resource is not compliant with the control, then it isn't provisioned. For more information, see the Controls reference guide in the AWS Control Tower documentation and see Proactive controls in *Implementing security controls on AWS*.

product lifecycle management (PLM)

The management of data and processes for a product throughout its entire lifecycle, from design, development, and launch, through growth and maturity, to decline and removal.

production environment

See environment.

programmable logic controller (PLC)

In manufacturing, a highly reliable, adaptable computer that monitors machines and automates manufacturing processes.

prompt chaining

Using the output of one LLM prompt as the input for the next prompt to generate better responses. This technique is used to break down a complex task into subtasks, or to iteratively refine or expand a preliminary response. It helps improve the accuracy and relevance of a model's responses and allows for more granular, personalized results.

pseudonymization

The process of replacing personal identifiers in a dataset with placeholder values. Pseudonymization can help protect personal privacy. Pseudonymized data is still considered to be personal data.

publish/subscribe (pub/sub)

A pattern that enables asynchronous communications among microservices to improve scalability and responsiveness. For example, in a microservices-based MES, a microservice can publish event messages to a channel that other microservices can subscribe to. The system can add new microservices without changing the publishing service.

# Q

query plan

A series of steps, like instructions, that are used to access the data in a SQL relational database system.

query plan regression

When a database service optimizer chooses a less optimal plan than it did before a given change to the database environment. This can be caused by changes to statistics, constraints, environment settings, query parameter bindings, and updates to the database engine.

# R

RACI matrix

See responsible, accountable, consulted, informed (RACI).

RAG

See Retrieval Augmented Generation.

ransomware

A malicious software that is designed to block access to a computer system or data until a payment is made.

RASCI matrix

See responsible, accountable, consulted, informed (RACI).

RCAC

See row and column access control.

read replica

A copy of a database that's used for read-only purposes. You can route queries to the read replica to reduce the load on your primary database.

re-architect

See 7 Rs.

recovery point objective (RPO)

The maximum acceptable amount of time since the last data recovery point. This determines what is considered an acceptable loss of data between the last recovery point and the interruption of service.

recovery time objective (RTO)

The maximum acceptable delay between the interruption of service and restoration of service.

refactor

See 7 Rs.

Region

A collection of AWS resources in a geographic area. Each AWS Region is isolated and independent of the others to provide fault tolerance, stability, and resilience. For more information, see Specify which AWS Regions your account can use.

regression

An ML technique that predicts a numeric value. For example, to solve the problem of "What price will this house sell for?" an ML model could use a linear regression model to predict a house's sale price based on known facts about the house (for example, the square footage).

rehost

See 7 Rs.

release

In a deployment process, the act of promoting changes to a production environment.

relocate

See 7 Rs.

replatform

See 7 Rs.

repurchase

See 7 Rs.

resiliency

An application's ability to resist or recover from disruptions. High availability and disaster recovery are common considerations when planning for resiliency in the AWS Cloud. For more information, see AWS Cloud Resilience.

resource-based policy

A policy attached to a resource, such as an Amazon S3 bucket, an endpoint, or an encryption key. This type of policy specifies which principals are allowed access, supported actions, and any other conditions that must be met.

responsible, accountable, consulted, informed (RACI) matrix

A matrix that defines the roles and responsibilities for all parties involved in migration activities and cloud operations. The matrix name is derived from the responsibility types defined in the matrix: responsible (R), accountable (A), consulted (C), and informed (I). The support (S) type is optional. If you include support, the matrix is called a *RASCI matrix*, and if you exclude it, it's called a *RACI matrix*.

responsive control

A security control that is designed to drive remediation of adverse events or deviations from your security baseline. For more information, see Responsive controls in *Implementing security controls on AWS*.

retain

See 7 Rs.

retire

See 7 Rs.

Retrieval Augmented Generation (RAG)

A generative AI technology in which an LLM references an authoritative data source that is outside of its training data sources before generating a response. For example, a RAG model might perform a semantic search of an organization's knowledge base or custom data. For more information, see What is RAG.

rotation

The process of periodically updating a secret to make it more difficult for an attacker to access the credentials.

row and column access control (RCAC)

The use of basic, flexible SQL expressions that have defined access rules. RCAC consists of row permissions and column masks.

RPO

See recovery point objective.

RTO

See recovery time objective.

runbook

A set of manual or automated procedures required to perform a specific task. These are typically built to streamline repetitive operations or procedures with high error rates.

# S

SAML 2.0

An open standard that many identity providers (IdPs) use. This feature enables federated single sign-on (SSO), so users can log into the AWS Management Console or call the AWS API operations without you having to create user in IAM for everyone in your organization. For more information about SAML 2.0-based federation, see About SAML 2.0-based federation in the IAM documentation.

SCADA

See supervisory control and data acquisition.

SCP

See service control policy.

secret

In AWS Secrets Manager, confidential or restricted information, such as a password or user credentials, that you store in encrypted form. It consists of the secret value and its metadata.

The secret value can be binary, a single string, or multiple strings. For more information, see What's in a Secrets Manager secret? in the Secrets Manager documentation.

security by design

A system engineering approach that takes security into account through the whole development process.

security control

A technical or administrative guardrail that prevents, detects, or reduces the ability of a threat actor to exploit a security vulnerability. There are four primary types of security controls: preventative, detective, responsive, and proactive.

security hardening

The process of reducing the attack surface to make it more resistant to attacks. This can include actions such as removing resources that are no longer needed, implementing the security best practice of granting least privilege, or deactivating unnecessary features in configuration files.

security information and event management (SIEM) system

Tools and services that combine security information management (SIM) and security event management (SEM) systems. A SIEM system collects, monitors, and analyzes data from servers, networks, devices, and other sources to detect threats and security breaches, and to generate alerts.

security response automation

A predefined and programmed action that is designed to automatically respond to or remediate a security event. These automations serve as detective or responsive security controls that help you implement AWS security best practices. Examples of automated response actions include modifying a VPC security group, patching an Amazon EC2 instance, or rotating credentials.

server-side encryption

Encryption of data at its destination, by the AWS service that receives it.

service control policy (SCP)

A policy that provides centralized control over permissions for all accounts in an organization in AWS Organizations. SCPs define guardrails or set limits on actions that an administrator can delegate to users or roles. You can use SCPs as allow lists or deny lists, to specify which services or actions are permitted or prohibited. For more information, see Service control policies in the AWS Organizations documentation.

service endpoint

The URL of the entry point for an AWS service. You can use the endpoint to connect programmatically to the target service. For more information, see AWS service endpoints in *AWS General Reference*.

service-level agreement (SLA)

An agreement that clarifies what an IT team promises to deliver to their customers, such as service uptime and performance.

service-level indicator (SLI)

A measurement of a performance aspect of a service, such as its error rate, availability, or throughput.

service-level objective (SLO)

A target metric that represents the health of a service, as measured by a service-level indicator.

shared responsibility model

A model describing the responsibility you share with AWS for cloud security and compliance. AWS is responsible for security *of* the cloud, whereas you are responsible for security *in* the cloud. For more information, see Shared responsibility model.

SIEM

See security information and event management system.

single point of failure (SPOF)

A failure in a single, critical component of an application that can disrupt the system.

SLA

See service-level agreement.

SLI

See service-level indicator.

SLO

See service-level objective.

split-and-seed model

A pattern for scaling and accelerating modernization projects. As new features and product releases are defined, the core team splits up to create new product teams. This helps scale your

organization's capabilities and services, improves developer productivity, and supports rapid innovation. For more information, see Phased approach to modernizing applications in the AWS Cloud.

SPOF

See single point of failure.

star schema

A database organizational structure that uses one large fact table to store transactional or measured data and uses one or more smaller dimensional tables to store data attributes. This structure is designed for use in a data warehouse or for business intelligence purposes.

strangler fig pattern

An approach to modernizing monolithic systems by incrementally rewriting and replacing system functionality until the legacy system can be decommissioned. This pattern uses the analogy of a fig vine that grows into an established tree and eventually overcomes and replaces its host. The pattern was introduced by Martin Fowler as a way to manage risk when rewriting monolithic systems. For an example of how to apply this pattern, see Modernizing legacy Microsoft ASP.NET (ASMX) web services incrementally by using containers and Amazon API Gateway.

subnet

A range of IP addresses in your VPC. A subnet must reside in a single Availability Zone.

supervisory control and data acquisition (SCADA)

In manufacturing, a system that uses hardware and software to monitor physical assets and production operations.

symmetric encryption

An encryption algorithm that uses the same key to encrypt and decrypt the data.

synthetic testing

Testing a system in a way that simulates user interactions to detect potential issues or to monitor performance. You can use Amazon CloudWatch Synthetics to create these tests.

system prompt

A technique for providing context, instructions, or guidelines to an LLM to direct its behavior. System prompts help set context and establish rules for interactions with users.

# T

tags

Key-value pairs that act as metadata for organizing your AWS resources. Tags can help you manage, identify, organize, search for, and filter resources. For more information, see Tagging your AWS resources.

target variable

The value that you are trying to predict in supervised ML. This is also referred to as an *outcome variable*. For example, in a manufacturing setting the target variable could be a product defect.

task list

A tool that is used to track progress through a runbook. A task list contains an overview of the runbook and a list of general tasks to be completed. For each general task, it includes the estimated amount of time required, the owner, and the progress.

test environment

See environment.

training

To provide data for your ML model to learn from. The training data must contain the correct answer. The learning algorithm finds patterns in the training data that map the input data attributes to the target (the answer that you want to predict). It outputs an ML model that captures these patterns. You can then use the ML model to make predictions on new data for which you don't know the target.

transit gateway

A network transit hub that you can use to interconnect your VPCs and on-premises networks. For more information, see What is a transit gateway in the AWS Transit Gateway documentation.

trunk-based workflow

An approach in which developers build and test features locally in a feature branch and then merge those changes into the main branch. The main branch is then built to the development, preproduction, and production environments, sequentially.

trusted access

Granting permissions to a service that you specify to perform tasks in your organization in AWS Organizations and in its accounts on your behalf. The trusted service creates a service-linked role in each account, when that role is needed, to perform management tasks for you. For more information, see Using AWS Organizations with other AWS services in the AWS Organizations documentation.

tuning

To change aspects of your training process to improve the ML model's accuracy. For example, you can train the ML model by generating a labeling set, adding labels, and then repeating these steps several times under different settings to optimize the model.

two-pizza team

A small DevOps team that you can feed with two pizzas. A two-pizza team size ensures the best possible opportunity for collaboration in software development.

## U

uncertainty

A concept that refers to imprecise, incomplete, or unknown information that can undermine the reliability of predictive ML models. There are two types of uncertainty: *Epistemic uncertainty* is caused by limited, incomplete data, whereas *aleatoric uncertainty* is caused by the noise and randomness inherent in the data. For more information, see the Quantifying uncertainty in deep learning systems guide.

undifferentiated tasks

Also known as *heavy lifting*, work that is necessary to create and operate an application but that doesn't provide direct value to the end user or provide competitive advantage. Examples of undifferentiated tasks include procurement, maintenance, and capacity planning.

upper environments

See environment.

# V

vacuuming

A database maintenance operation that involves cleaning up after incremental updates to reclaim storage and improve performance.

version control

Processes and tools that track changes, such as changes to source code in a repository.

VPC peering

A connection between two VPCs that allows you to route traffic by using private IP addresses. For more information, see What is VPC peering in the Amazon VPC documentation.

vulnerability

A software or hardware flaw that compromises the security of the system.

# W

warm cache

A buffer cache that contains current, relevant data that is frequently accessed. The database instance can read from the buffer cache, which is faster than reading from the main memory or disk.

warm data

Data that is infrequently accessed. When querying this kind of data, moderately slow queries are typically acceptable.

window function

A SQL function that performs a calculation on a group of rows that relate in some way to the current record. Window functions are useful for processing tasks, such as calculating a moving average or accessing the value of rows based on the relative position of the current row.

workload

A collection of resources and code that delivers business value, such as a customer-facing application or backend process.

workstream

Functional groups in a migration project that are responsible for a specific set of tasks. Each workstream is independent but supports the other workstreams in the project. For example, the portfolio workstream is responsible for prioritizing applications, wave planning, and collecting migration metadata. The portfolio workstream delivers these assets to the migration workstream, which then migrates the servers and applications.

WORM

See write once, read many.

WQF

See AWS Workload Qualification Framework.

write once, read many (WORM)

A storage model that writes data a single time and prevents the data from being deleted or modified. Authorized users can read the data as many times as needed, but they cannot change it. This data storage infrastructure is considered immutable.

# Z

zero-day exploit

An attack, typically malware, that takes advantage of a zero-day vulnerability.

zero-day vulnerability

An unmitigated flaw or vulnerability in a production system. Threat actors can use this type of vulnerability to attack the system. Developers frequently become aware of the vulnerability as a result of the attack.

zero-shot prompting

Providing an LLM with instructions for performing a task but no examples (*shots*) that can help guide it. The LLM must use its pre-trained knowledge to handle the task. The effectiveness of zero-shot prompting depends on the complexity of the task and the quality of the prompt. See also few-shot prompting.

zombie application

An application that has an average CPU and memory usage below 5 percent. In a migration project, it is common to retire these applications.