User Guide

AWS Account Management

AWS Account Management: User Guide

Table of Contents

What is an AWS account?

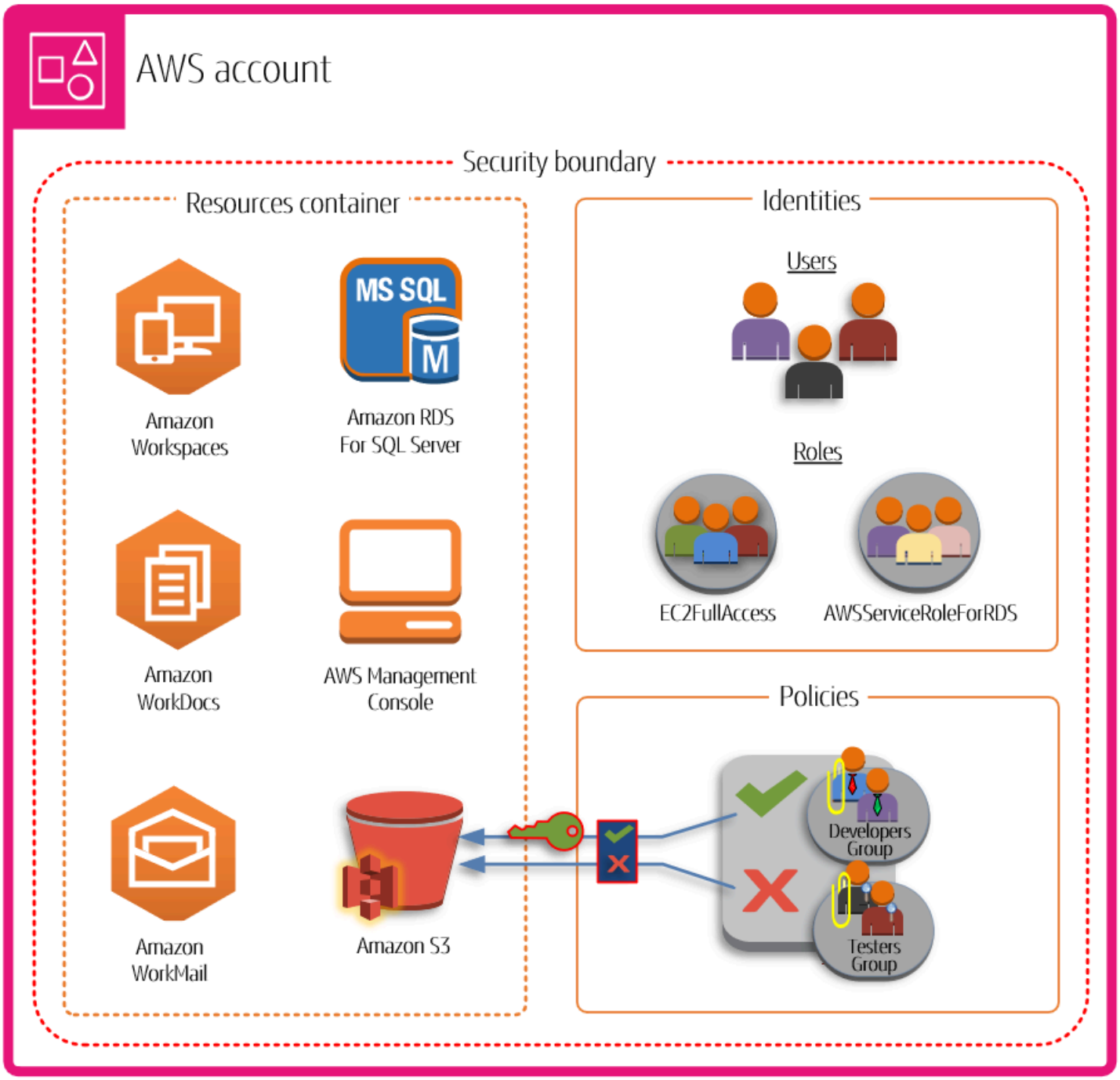
An AWS account represents a formal business relationship you establish with AWS. You create and manage your AWS resources in an AWS account, and your account provides identity management capabilities for access and billing. Each AWS account has a unique ID which differentiates it from other AWS accounts.

When you sign up for AWS, you can choose between Sign up for AWS (new) and Sign up for AWS (advanced). Depending on your choice, you either have access to AWS accounts or projects. Projects contain AWS accounts and the settings for sharing with other collaborators. For more information, see Compare sign-up options.

Your cloud resources and data are contained in an AWS account. An account acts as an identity and access management isolation boundary. When you need to share resources and data between two accounts, you must explicitly allow this access. By default, no access is allowed between accounts. For example, if you designate different accounts to contain your production and non-production resources and data, no access is allowed between those environments by default.

AWS accounts are also a fundamental part of accessing AWS services. As shown in the following illustration, an AWS account serves two primary functions:

- **Resources container** – An AWS account is the basic container for all the AWS resources you create as an AWS customer. For example, an Amazon Simple Storage Service (Amazon S3) bucket, an Amazon Relational Database Service (Amazon RDS) database, and an Amazon Elastic Compute Cloud (Amazon EC2) instance are all resources. Every resource is uniquely identified by an Amazon Resource Name (ARN) that includes the account ID of the account that contains, or owns, the resource.
- **Security boundary** – An AWS account is also the basic security boundary for your AWS resources. Resources that you create in your account are available to users who have credentials for your account. Among the key resources you can create in your account are *identities*, such as users and roles. Identities have credentials that someone can use to sign in (*authenticate*) to AWS. Identities also have permission policies that specify what a user can do (*authorization*) with the resources in the account.



Using multiple AWS accounts is a best practice for scaling your environment, as it provides a natural billing boundary for costs, isolates resources for security, gives flexibility for individuals and teams, in addition to being adaptable for new business processes. For more information, see Benefits of using multiple AWS accounts.

Features of an AWS account

AWS accounts include the following core features:

- **Monitor and control costs** – An account is the default means by which AWS costs are allocated. Because of this fact, using different accounts for different business units and groups of workloads can help you more easily track, control, forecast, budget, and report your cloud expenditures. In addition to cost reporting at the account level, AWS also has built-in support to consolidate and report costs across your entire set of accounts should you choose to use AWS Organizations at some point. You can also use AWS Service Quotas to help protect you from unexpected excessive provisioning of AWS resources and malicious actions that could dramatically impact your AWS costs.
- **Unit of isolation** – An AWS account provides security, access, and billing boundaries for your AWS resources that can help you achieve resource autonomy and isolation. By design, all resources provisioned within an account are logically isolated from resources provisioned in other accounts, even within your own AWS environment. This isolation boundary provides you with a way to limit the risks of an application-related issue, misconfiguration, or malicious actions. If an issue occurs within one account, impacts to workloads contained in other accounts can be either reduced or eliminated.
- **Mirror your business workloads** – Use multiple accounts to group workloads with a common business purpose in distinct accounts. As a result, you can align the ownership and decision making with those accounts and avoid dependencies and conflicts with how workloads in other accounts are secured and managed. Depending on your overall business model, you might choose to isolate distinct business units or subsidiaries in different accounts. This approach might also ease divestment of those units over time.

Are you a first-time AWS user?

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

If you're a first-time user of AWS, your first step is to sign up for AWS. You can either create an AWS account where you have complete control over account configuration and where your content is

stored, which is required for regulated workloads such as HIPAA or FedRAMP, or you can create AWS accounts with preconfigured defaults. To learn which sign up is right for you, see Compare sign-up options.

For step-by-step instructions on how to set up a new account where you have complete control over the initial setup, see Sign up for AWS (advanced).

For step-by-step instructions on how to set up a new account where AWS creates accounts with preconfigured defaults, see Sign up for AWS (new).

Support for AWS Account Management

You can post feedback and questions using the AWS Account Management support forum. For general information about AWS forums, see AWS re:Post.

If you can't find the answers you are looking for on AWS re:Post, you can create an account or billing related support case using the AWS Management Console. For more information, see Example: Create a support case for account and billing.

Other AWS resources

- **AWS Training and Courses** – Links to role-based and specialty courses as well as self-paced labs to help sharpen your AWS skills and gain practical experience.
- **AWS Developer Tools** – Links to developer tools and resources that provide documentation, code examples, release notes, and other information to help you build innovative applications with AWS.
- **AWS Support Center** – The hub for creating and managing your AWS Support cases. Also includes links to other helpful resources, such as forums, technical FAQs, service health status, and AWS Trusted Advisor.
- **AWS Support** – The primary webpage for information about AWS Support, a one-on-one, fast-response support channel to help you build and run applications in the cloud.
- **Contact Us** – A central contact point for inquiries concerning AWS billing, account, events, abuse, and other issues.
- **AWS Site Terms** – Detailed information about our copyright and trademark; your account, license, and site access; and other topics.

Compare sign-up options

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you sign up for AWS, you unlock the world's most comprehensive and broadly adopted cloud. We offer the greatest choice of innovative cloud and AI capabilities and expertise, on the most extensive global infrastructure, with industry-leading security, reliability, and performance.

There are two ways to sign up for AWS. If you want complete control over account configuration and where your content is stored, or if you have regulated workloads such as HIPAA, FedRAMP, use Sign up for AWS (advanced). If you are a new user and want to use a login you already own to sign in to AWS and create an AWS environment with preconfigured defaults in select AWS Regions, use Sign up for AWS (new).

Supported AWS services for Sign up for AWS (new)

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you use Sign up for AWS (new), you get access to the Free Tier. Free Tier provides access to select AWS services. You can upgrade your account to a Paid Plan to access additional AWS services and features. To learn how to upgrade your account, see Upgrade your account in AWS Settings.

Not every AWS service is available for Sign up for AWS (new). If a service is not listed in either of the following lists or has an unsupported feature, you must create an account by using Sign up for AWS (advanced) or Activate advanced AWS features.

AWS services included in the Free Tier of our new AWS experience

The following AWS services are included in the Free Tier of Sign up for AWS (new):

- Amazon API Gateway
- Amazon Athena
- Amazon AppFlow
  - Not supported in Europe (Stockholm).
- AWS App2Container
- Amazon Augmented AI
  - Not supported in Europe (Stockholm).
- Amazon Bedrock
  - Global cross-Region inference and Geographic cross-Region inference are not supported.
- Amazon Bedrock Agent Core
- Amazon CloudFront
  - Lambda@Edge is not supported.
- Amazon CloudWatch
  - Cross-account observability and cross-Region dashboards are not supported.
- Amazon CloudWatch Logs
- Amazon CodeGuru Profiler
- Amazon Cognito
- Amazon Cognito Sync
- Amazon DynamoDB
  - Global Tables (multi-Region replication) are not supported.
- Amazon DataZone
- Amazon Elastic Container Registry (ECR)
  - Cross-Region replication is not supported.
- EC2 Image Builder
- Amazon EKS Anywhere
- Amazon Elastic Block Store (EBS)
- Amazon Elastic Compute Cloud (EC2)
  - Cross-Region AMI copy and cross-Region snapshot copy are not supported.
  - Purchase Capacity Block and Purchase Capacity Block Extension are not supported.
- Amazon Elastic Container Registry Public

- Amazon Elastic Container Service
- Amazon Elastic Kubernetes Service
- Amazon Elastic File System
  - Cross-Region replication is not supported.
- Amazon ElastiCache
  - Global Datastores (cross-Region replication) are not supported.
- Amazon EventBridge
- Amazon Lex
  - Global Resiliency (bot replication across Regions) is not supported.
- Amazon Location Service
- Amazon Managed Service for Prometheus
  - Cross-Region projects are not supported.
- Amazon Managed Workflows for Apache Airflow
- Amazon MQ
  - Cross-Region data replication (CRDR) is not supported.
- Amazon OpenSearch Service
  - Cross-cluster replication across Regions is not supported.
- Amazon Polly
- Amazon Quick
  - Only Amazon Quick Free or Quick Plus are supported. For more information, see Amazon Quick pricing.
- Amazon Rekognition
  - Not supported in Europe (Stockholm).
- Amazon Relational Database Service
  - Cross-Region read replicas and Aurora Global Databases are not supported.
- Amazon Route 53
  - Cross-Region routing policies (geolocation, latency-based, failover) are not supported.
- Amazon SageMaker
  - Cross-region deployment and cross-region inference are not supported.
- Amazon SageMaker AI

- Amazon SageMaker Unified Studio
- Amazon Simple Email Service
  - Global endpoints (multi-region sending with failover) are not supported.
- Amazon Simple Notification Service
- Amazon Simple Queue Service
- Amazon Simple Storage Service
  - Cross-Region Replication (CRR) and Multi-Region Access Points are not supported.
  - Amazon S3 Access Grants is not supported.
- Amazon Simple Workflow Service
- Amazon Verified Permissions
- Amazon Virtual Private Cloud
  - Cross-Region VPC peering and Transit Gateway inter-Region peering are not supported.
- Amazon Aurora
- Amazon Aurora DSQL
  - Multi-Region active-active clusters are not supported.
- AWS Amplify
- AWS AppConfig
- AWS AppSync
- Application Auto Scaling
- AWS Transform MGN
- AWS Artifact
  - (HIPAA) AWS Business Associate Addendum, (HIPAA) AWS Organizations Business Associate Addendum, AWS SEC Rule 17a-4 Addendum, and AWS SEC Rule 18a-6 Addendum are not supported.
- AWS B2B Data Interchange
- AWS Backint agent
- AWS Backup
  - Cross-Region backup copies are not supported.
- AWS Batch
- AWS Billing and Cost Management

Use AWS Settings to set up your billing, create spend limits to control your costs, and retrieve and pay invoices. You use Billing and Cost Management to create budgets, analyze, and optimize your costs. For more information, see Using Billing and Cost Management console with our new AWS experience.

- AWS Budgets
- AWS Certificate Manager
  - AWS Certificate Manager exportable public certificates are not supported.
- AWS Cloud Control API
- AWS Cloud Map
- AWS Cloud9
- AWS CloudFormation
  - StackSets (multi-account, multi-Region deployments) are not supported.
- AWS CloudShell
- AWS CloudTrail
  - Multi-Region trails and organization trails are not supported.
- AWS CodeArtifact
- AWS CodeCommit
- AWS CodeDeploy
- AWS CodePipeline
  - Cross-Region actions are not supported.
- AWS Compute Optimizer
- AWS Config
  - Multi-Account Multi-Region Data Aggregation is not supported.
- AWS Console Mobile Application
- AWS Cost Explorer
- AWS Data Pipeline
- AWS Data Transfer
- AWS DataSync
  - Cross-Region transfers are not supported.
- AWS Directory Service

- AWS Elastic Beanstalk
- AWS End User Messaging Social
- AWS Entity Resolution
  - Cross-Region namespaces are not supported.
- AWS Glue
- AWS Health
- AWS Identity and Access Management
- AWS IoT
  - AWS Greengrass is not supported.
- AWS IoT Device Defender
- AWS IoT Device Management
- AWS IoT SiteWise
  - Not supported in Europe (Stockholm).
- AWS IoT TwinMaker
- AWS Key Management Service
  - Multi-Region keys are not supported.
- AWS Lake Formation
- AWS Lambda
  - Lambda@Edge is not supported.
- AWS Launch Wizard
- AWS License Manager
- AWS Management Console
- AWS Marketplace
  - Bedrock and Free only.
- AWS Network Manager
  - Global network management across Regions is not supported.
- AWS Payment Cryptography
- AWS PrivateLink
- AWS Resource Access Manager (RAM)
- AWS Resource Explorer

  - Cross-Region search via aggregator index is not supported.
- AWS Resource Groups
- Route 53 Amazon Application Recovery Controller (ARC)
- AWS SDKs & Tools
- AWS Secrets Manager
  - Cross-Region secret replication is not supported.
- AWS Security Token Service
- AWS Serverless Application Repository
- Service Quotas
- AWS Signer
- AWS Step Functions
- AWS Support
- AWS Systems Manager
  - Multi-account multi-Region resource data syncs are not supported.
- AWS Systems Manager for SAP
- AWS User Notifications
  - User-configured notifications and Managed notifications configuration operations are not supported. For more information, see What is AWS User Notifications?
- AWS WAF
  - AWS Shield is not supported.
- AWS Well-Architected Tool
- AWS X-Ray
- AWS CodeBuild
- Elastic Load Balancing
- AWS SQL Workbench
- Amazon OpenSearch Service Serverless
  - Cross-Region data access is not supported.
- Amazon Timestream

To see the policy that controls access to the Free Tier services, see Service control policies for the Free Tier for projects.

AWS services included in the Paid Plan of our new AWS experience

The following services are available for Sign up for AWS (new) if you have a Paid Plan. For more information, see Upgrade your account in AWS Settings.

- Amazon DevOps Guru
- Amazon DocumentDB (with MongoDB compatibility)
  - Global clusters (cross-Region) are not supported.
- Amazon EMR
  - Cross-region processing for Spark Upgrade Agent is not supported.
- Amazon Forecast
- Amazon Glacier
- Amazon Kinesis
  - Cross-Region stream sharing is not supported.
- Amazon Kinesis Data Streams
- Amazon Data Firehose
- Amazon Kinesis Video Streams
  - Not supported in Europe (Stockholm).
- Amazon Managed Blockchain
  - Not supported in Europe (Stockholm) / US East (Ohio) / Asia Pacific (Sydney).
- Amazon Managed Streaming for Apache Kafka
  - Cross-Region replication (MSK Replicator) is not supported.
- Amazon MemoryDB
  - Multi-Region replication is not supported.
- Amazon Neptune
- Amazon Personalize
  - Not supported in Europe (Stockholm).
- Amazon Redshift
- S3 Glacier Deep Archive
- Amazon Textract
  - Not supported in Europe (Stockholm).

- Amazon Transcribe
- Amazon Translate
- AWS AppFabric
- AWS Data Exchange
- AWS End User Messaging SMS
- AWS Fault Injection Service
- AWS Firewall Manager
- AWS Network Firewall
- AWS Security Agent
- AWS Storage Gateway
- DevOps Agent

To see the policy that controls access to the Paid Plan services, see Service control policies for the Paid Plan for projects.

AWS services not supported for our new AWS experience

The following AWS services are not supported for our new AWS experience, unless you Activate advanced AWS features. These AWS services aren't necessary for new developers or small teams. Some of these services are automatically turned on for your account and managed by AWS. For more information, see Compare features for sign-up options.

- Alexa for Business
- Alexa Top Sites
- Alexa Web Information Service
- Amazon Braket
- Amazon Chime
- Amazon Chime Business Calling a service sold by AMCS LLC
- Amazon Chime Call Me
- Amazon Chime Call Me a service sold by AMCS LLC
- Amazon Chime Dial In a service sold by AMCS LLC
- Amazon Chime Dialin

- Amazon Chime Features
- Amazon Chime SDK
- Amazon Chime Services
- Amazon Chime SMS
- Amazon Chime Voice Connector a service sold by AMCS LLC
- Amazon Cloud Directory
- Amazon CloudSearch
- Amazon CodeWhisperer
- Amazon Connect Customer
- Amazon Connect Customer Customer Profiles
- Amazon Connect Customer Voice ID
- AWS Deadline Cloud
- Amazon Detective
- Amazon Elastic Inference
- Amazon Elastic Transcoder
- Amazon FinSpace
- Amazon Flexible Fleet
- Amazon Fraud Detector
- Amazon FSx
- Amazon GameLift Servers
- Amazon GameLift Streams
- Amazon GameSparks
- Amazon GuardDuty
- AWS HealthOmics
- Amazon Inspector Classic
- Amazon Interactive Video Service
- Amazon IVS Chat
- Amazon Kendra
- Amazon Keyspaces (for Apache Cassandra)
- Kiro

- Amazon Lightsail
- Amazon Macie
- Amazon Managed Grafana
- Amazon Mechanical Turk
- Amazon Mechanical Turk Worker Rewards
- Amazon Mobile Analytics
- Amazon Monitron
- Amazon Nimble Studio
- Amazon OpenSearch Service Ingestion Service
- Amazon Q
- Amazon Q Business
- Amazon Q Developer
- Amazon Quantum Ledger Database (Amazon QLDB)
- Amazon S3 on Outposts
- Amazon SageMaker Ground Truth
- Amazon Security Lake
- Amazon Sumerian
- Amazon WorkDocs
- Amazon WorkSpaces
- Amazon WorkSpaces Instances
- Amazon WorkSpaces Applications (Amazon WorkSpaces Applications)
- Amazon WorkSpaces Thin Client
- Amazon WorkSpaces Web
- Amazon WorkLink
- Amazon WorkMail
- AWS Application Cost Profiler
- AWS Clean Rooms
- AWS App Studio
- AWS Application Discovery Service
- AWS Audit Manager

- AWS Billing Conductor
- AWS CloudHSM
- AWS Client VPN
- AWS Cloud WAN
- AWS Control Tower
- AWS Database Migration Service
- AWS DeepRacer
- AWS Device Farm
- AWS Direct Connect
- AWS Elastic Disaster Recovery
- AWS Elemental Live
- AWS Elemental MediaConnect
- AWS Elemental MediaConvert
- AWS Elemental MediaLive
- AWS Elemental MediaPackage
- AWS Elemental MediaStore
- AWS Elemental MediaTailor
- AWS Global Accelerator
- AWS Ground Station
- AWS HealthImaging
- AWS IAM Identity Center
- AWS Identity and Access Management Access Analyzer
- AWS Interconnect
- AWS IoT Events
- AWS IoT FleetWise
- AWS IoT RoboRunner
- AWS Mainframe Modernization
- AWS Migration Hub
- AWS Migration Hub Refactor Spaces
- AWS Modular Data Center

- AWS OpsWorks
- AWS Organizations
- AWS Outposts
- AWS Panorama
- AWS Parallel Computing Service
- AWS Partner Network
- AWS Pricing Calculator
- AWS Private Certificate Authority
- AWS Proton
- AWS re:Post Private
- AWS Resilience Hub
- AWS RoboMaker
- AWS RTB Fabric
- AWS Security Hub
- AWS Security Incident Response
- AWS Service Catalog
- AWS Shield
- AWS SimSpace Weaver
- AWS Snowball
- AWS Supply Chain
- AWS Telco Network Builder
- AWS Transform
- AWS Transfer Family
- AWS Transit Gateway
- AWS Trusted Advisor
- AWS Verified Access
- Site-to-Site VPN
- Amazon VPC Lattice
- AWS Wickr
- Amazon CodeCatalyst

- Amazon Comprehend Medical
- Conferencing Telecommunications (service sold by AMCS LLC)
- Contact Center Telecommunications (service sold by AMCS LLC)
- Contact Center Telecommunications Korea
- Contact Center Telecommunications South Africa
- DynamoDB Accelerator
- Amazon Elastic VMware Service
- Oracle Database@AWS
- Red Hat OpenShift Service on AWS
- AWS App Runner

Considerations for choosing how to sign up for AWS

Consider the following before choosing your sign-up method. You can activate all AWS features for your account if necessary.

- If you have specific product requirements such as regulated workloads that include HIPAA or FedRAMP, do not use our new AWS experience.
- Your content will be stored in a designated Region which is selected based on your contact information.
- The new AWS experience makes it easier for you to start building without creating AWS permissions. If you need fine-grained control over your users or role-based permissions, do not use Sign up for AWS (new).
- If you need account-specific payment configurations set up by AWS, also known as net terms billing, do not use Sign up for AWS (new).
- If you need access to the full set of AWS services, do not use Sign up for AWS (new). For more information, see Supported AWS services for Sign up for AWS (new).

Compare features for sign-up options

The following table compares AWS wide requirements that are either supported for Sign up for AWS (new) or Sign up for AWS (advanced). If a feature is not listed in this table or described in Compare access management, it is available. You can activate advanced features for your account

if you need any of the unsupported features. For more information, see Activate advanced AWS features.

Feature	Sign up for AWS (new)	Sign up for AWS (advanced)
Join AWS Partner Network	No	Yes
Create a Professional Services contract	No	Yes
Enroll in an Enterprise Agreement with AWS	No	Yes
Purchase an AWS Skill Builder Team subscription	No	Yes
Work with an AWS Training Partner	No	Yes
Earn extra credits from AWS Activate	No	Yes
Use AWS Marketplace	No	Yes
Designate your AWS account as HIPAA or SEC compliant	No	Yes
Select a specific AWS Region to create your Regional resources	No. For more information, see AWS Regions for your projects.	Yes
Access an opt-in Region	No	Yes
IAM Access Analyzer	No	Yes
Sign-in access policies	No	Yes
Create spend limits for your accounts	Yes	No
Savings plans	No	Yes
Automatic opt-in to Cost Optimization Hub for the Paid Plan	Yes	No

Feature	Sign up for AWS (new)	Sign up for AWS (advanced)
Automatic opt-in to AWS Organizations	Yes	No
Automatic opt-in to IAM Identity Center	Yes	No
Automatically opt in to IAM role manager	Yes	No
Console account colors	No	Yes
Use the AWS Console Mobile Application	No	Yes
Disable multi-session console support	No	Yes. If you sign in again using a Builder ID, multi-session support will be enabled automatic ally.

Compare access management

When you use our new AWS experience, AWS creates preconfigured defaults to help you get started building. Your AWS resources are organized in projects. A project contains an AWS account and settings for sharing with other collaborators. If you use the paid plan, your project can also have a set monthly pre-tax cost, called a spend limit. All of the projects you own make up your organization, which you control through your management account in AWS Settings. This organization is the same as an AWS Organization; however, AWS manages aspects of this organization on your behalf.

AWS manages the following elements of your AWS Organization:

- **Organization management policies.** When you use our new AWS experience, AWS manages the organization management policies including the resource control policies (RCPs) and the service control policies (SCPs). If you want to create your own, use Sign up for AWS (advanced).
- **Human access roles.** A human access role is any role that allows a human to have specific permissions. For example, a human access role could let the user tester have read only access to Amazon Bedrock. While you can create IAM roles and IAM users, we recommend only using them

when necessary. You do not create a root user. AWS provides you and your team members with admin access to your projects. You can add and remove team members to your project. For more information, see Invite team members to collaborate in AWS Settings.

Every project has a managed security implementation that makes it easy to build with confidence. By default, any application or code in your project has access to all AWS resources in that same project. You do not need to perform additional access configuration.

AWS resources in different projects cannot access each other, unless you turn on cross-project access for certain resources. For instance, you can create a Lambda function in one project and then invoke it using an API in a different project. This is described as a cross-account integration. For more information, see Cross-account Lambda integrations. As long as both projects are owned by the same management account, this is supported.

By default, your project does not provide external access to any resources. You must configure AWS resources to allow for public access. You can do the following:

- Create a public S3 bucket to host a simple static website or files. For more information, see Getting started with Amazon S3.
- Create an API Gateway API to act as the front door of your application. For more information, see Build an HTTP API.
- Create an EC2 instance with a public IP address to allow for communication between your instances and the Internet. For more information, see Launch your first EC2 instance.
- Create a CloudFront distribution to deliver your static and dynamic web content, such as .html, .css, .js, and image files, to your users. For more information, see Getting started with CloudFront.
- Create a Lambda function URL to provide a simple, direct publicly accessible HTTP endpoint for a Lambda function. Make sure your Lambda function has the correct permissions. For more information, see Creating a function URL and Security and auth model for Lambda function URLs.
- Use a combination of these or other AWS mechanisms.

In addition, when you activate advanced features, you have access to even more mechanisms to provide external access to your resources.

Manage your AWS accounts

You manage your AWS accounts based on the type of AWS you're using. The following table shows where to find the documentation for each management task.

Task	Sign up for AWS (new)	Sign up for AWS (advanced)
Sign up	Sign up for AWS (new)	Sign up for AWS (advanced)
Create an account	Create a project in AWS Settings	Sign up for AWS (advanced)
Close an account	Close a project in AWS Settings	Close an AWS account
Update your password	Update your password in AWS Settings	Update root user password
Register MFA devices	Manage multi-factor authentication (MFA) in AWS Settings	Step 3: (Recommended) Set up the AWS MCP Server
Change your email	Change your email address in AWS Settings	Update the root user email address
Change your name	Manage your name or nickname in AWS Settings	Update your AWS account name
Update contact information	Edit your contact information in AWS Settings	Update the primary contact for your AWS account

Task	Sign up for AWS (new)	Sign up for AWS (advanced)
Update alternate contacts	Not applicable	Update the alternate contacts for your AWS account
Manage billing	Manage your payment method in AWS Settings	Update billing for your AWS account
Manage tax information	Manage tax registration in AWS Settings	Manage your payments
Enable or disable AWS Regions	Not applicable. You cannot modify your AWS Region.	Enable or disable AWS Regions in your account
View account identifiers	Not applicable	View AWS account identifiers
Create an account alias	Not applicable	Create an AWS account alias
Upgrade your account	Upgrade your account in AWS Settings	Free Tier plans
Create a spend limit	Create a spend limit in AWS Settings	Not applicable
Update a spend limit	Update a spend limit in AWS Settings	Not applicable
Remove a spend limit	Remove a spend limit in AWS Settings	Not applicable

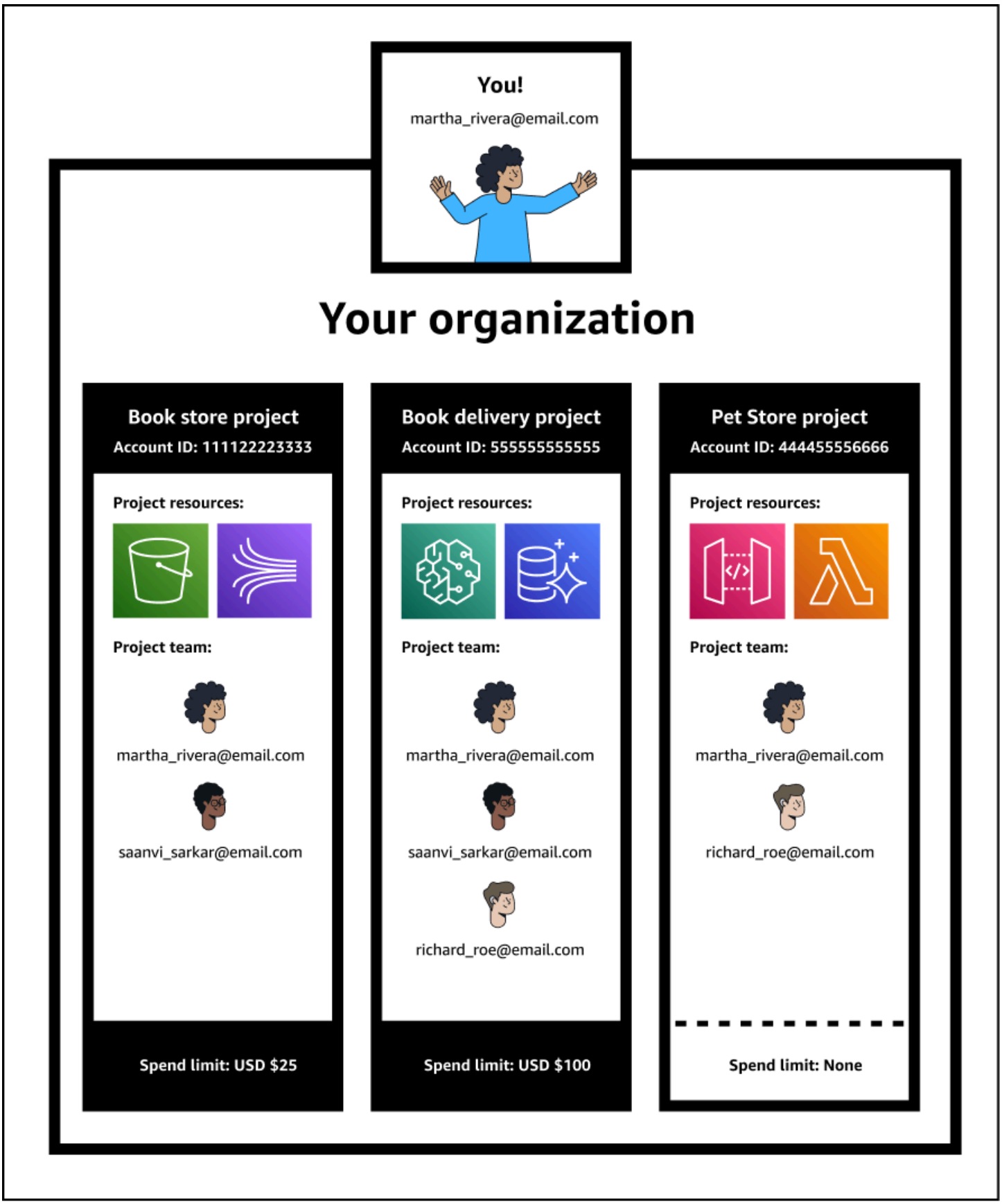
Task	Sign up for AWS (new)	Sign up for AWS (advanced)
Invite team members	Invite team members to collaborate in AWS Settings	IAM Identities
Switch between projects	Switch between projects	Multi-session support
Connect an AI coding tool	Connect an AI coding tool	Not applicable
Delete all sessions	Delete all sessions in AWS Settings	Not applicable
Opt out of AI data use	Opt out of use of your content for service improvement in AWS Settings	AI services opt-out policies
Request your data	Request your AWS Builder ID data	Request your data
Delete your AWS profile	Close your account in AWS Settings	Close an AWS account
Manage accounts in India	Manage accounts in India in AWS Settings	Manage accounts in India
Create an administrator user	Not applicable	Sign up for AWS (advanced)
Plan your account governance structure	Not applicable	Plan your AWS account governance structure

Sign up for AWS (new)

Warning

We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you use Sign up for AWS (new), you create a preconfigured AWS environment that lets you use a login that you already own like Google or GitHub to access your AWS resources. Your resources are organized in projects. A project contains an AWS account and settings for sharing with other collaborators. If you use the paid plan, your project can also have a set monthly pre-tax cost, called a spend limit. All of the projects you own make up your organization. You manage your organization, access your projects, and add collaborators in AWS Settings.


You!
martha_rivera@email.com
Your organization
Book store project
Account ID: 111122223333
Project resources:
Project team:
martha_rivera@email.com
saanvi_sarkar@email.com
Spend limit: USD $25
Book delivery project
Account ID: 555555555555
Project resources:
Project team:
martha_rivera@email.com
saanvi_sarkar@email.com
richard_roe@email.com
Spend limit: USD $100
Pet Store project
Account ID: 444455556666
Project resources:
Project team:
martha_rivera@email.com
richard_roe@email.com
Spend limit: None

When you sign up for AWS using Sign up for AWS (new), AWS also creates a AWS Builder ID account, which is a personal profile that represents you as an individual. You can only have one AWS Builder ID for each email address. If you've created a AWS Builder ID but have not created an AWS account, you might have some settings already made for you and you'll have to modify those later.

For more information about Sign up for AWS (new), see the following documentation:

- Compare sign-up options
- Supported AWS services for Sign up for AWS (new)

Sign up using Sign up for AWS (new)

When you sign up for AWS, we might request payment information. Our verification process holds USD $1 (or equivalent) for 3-5 days to verify your account and prevent fraud. You will not incur charges until you upgrade to a paid plan. You can manage your payment method or close your accounts at any time through AWS Settings. In addition, when you sign up for AWS, you might be ineligible for the Free Tier. In that case, you'll need to provide your payment information, and you'll be placed on the Paid Plan. When you're on the Paid Plan, you can set spend limits to operate within a predictable and controlled budget.

Sign up for AWS (new) supports using Google, GitHub, Apple, or Amazon for your account. If you're already signed into a Google account in your browser, that account will be used when you sign up for AWS.

To sign up for AWS using Sign up for AWS (new)

1. Go to aws.amazon.com and choose **Create account**.
2. On the banner explaining the differences between Sign up for AWS (new) and Sign up for AWS (advanced), choose **Sign up for AWS**.
3. Choose a login such as Google, Apple, GitHub, or Amazon.
4. You'll be directed to the login for your provider. After you log in, you'll be redirected to AWS sign up.
5. Enter your name and choose **Continue**.

   This name is shared with collaborators when you give them access to your projects.

6. Verify the email associated with your login by entering a verification code. When you've verified your email, choose **Continue**.

   This email is shared with collaborators when you give them access to your projects.

7. Enter a strong password for your account and choose **Continue**.

   AWS requires that your password meet the following conditions:

   - It must have a minimum of 8 characters and a maximum of 128 characters.
   - It must include a minimum of three of the following mix of character types: uppercase, lowercase, numbers, and ! @ # $ % ^ & * () <> [] {} | _+-= symbols.
   - It must not be identical to your AWS account name or email address.

8. Enter your contact details and choose **Continue**.
9. AWS will provision your first project in one of three AWS Regions with a new name. For more information, see AWS Regions for your projects. You can change the name at any time.

You can now access the AWS Management Console. To access AWS Settings, choose **Manage projects** from the AWS Management Console. In AWS Settings, you can invite collaborators, create new projects, and modify your billing and other settings.

Next steps

After you sign up for AWS, you can connect an AI coding tool, add team members to collaborate, and if you have a paid plan, create a spend limit.

- Connect an AI coding tool
- Invite team members to collaborate in AWS Settings
- Create a spend limit in AWS Settings

Use AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

AWS Settings is where you manage and access your projects. AWS Settings is available when you sign up using Sign up for AWS (new). You can use AWS Settings to do the following:

- Create and access projects you own. For more information, see Create a project in AWS Settings.
- Access projects that have been shared with you.
- Invite team members to collaborate on your projects. For more information, see Invite team members to collaborate in AWS Settings.
- Switch between projects. For more information, see Switch between projects.
- Manage your payment method. For more information, see Manage your payment method in AWS Settings.
- Upgrade from the free tier to a paid plan. For more information, see Upgrade your account in AWS Settings.
- Manage spend limits for your projects. For more information, see Create a spend limit in AWS Settings.
- View your billing information. For more information, see View your billing information in AWS Settings.
- Manage your profile information. For more information, see Manage your name or nickname in AWS Settings, Change your email address in AWS Settings, and Edit your contact information in AWS Settings.
- Manage your security settings and sign-in preferences. For more information, see Update your password in AWS Settings and Manage multi-factor authentication (MFA) in AWS Settings.

When you manage your profile information or your security settings, you'll be redirected to the AWS Builder ID hub. Your AWS Builder ID is created when you use Sign up for AWS (new). For more information, see What is AWS Builder ID?.

Connect an AI coding tool

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can provide your AI coding tool with access to your project and install the Agent Toolkit, which bundles the AWS MCP server configuration and a curated set of agent skills. This lets your AI coding tool create and manage AWS resources on your behalf.

Considerations for connecting your coding tool to your project

- You can connect AI coding tools to a project you own or a project that has been shared with you.
- When an AI coding tool or local agent creates AWS resources, you are still responsible for them. This includes the proper security of your resources as defined by the shared responsibility model and any charges that might occur from these resources, whether they are accounted for in the Free Plan or charged on the Paid Plan.
- You are responsible for any actions that your AI coding tool takes on your behalf in your projects.
- To view a resource that your agent created, go to the service console for that service.
- The credentials you generate are for use by the AWS CLI, AWS Tools for PowerShell, and AWS SDKs. External tools may not be supported. The access to your resources is valid for 12 hours. Use the following command to renew your credentials:

  ```
  aws login --profile profile-name
  ```

  Replace *profile-name* with the profile name you chose during initial setup. The command will automatically open your default browser. You'll need to choose your project name to finish the login process. AWS can renew your credentials for 90 days. After 90 days, you'll need to grant access in a browser window again.

Connect your AI coding tool to your project

Each project has its own set of credentials that AI coding tools can use to create and manage AWS resources on your behalf.

AWS Management Console

To connect your AI coding tool to your project using the AWS console

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose the project you want to access.
3. In the navigation bar on the upper right, choose **Create and manage cloud infrastructure**.

4. Choose **Use your AWS credentials with AI coding tools**.
5. Choose **Copy agent prompt** and enter it in your coding agent.

   This creates a prompt that shares a steering file with your AI coding agent. The steering file accesses a script from AWS that sets up the connection between your project credentials and your AI coding agent.

   You'll enter a profile name for your project. This profile name is how your AI coding tool accesses your resources.

Use your terminal

Run the following command to download and follow the setup instructions:

```
curl -fsSL https://github.com/aws/agent-toolkit-for-aws/blob/main/setup-
instructions/setup.md
```

When you use our new AWS experience, your agent will install the AWS MCP server, and you get access to agent skills that can be used with your projects. You'll also get a rule file with guidance tuned for this experience. The rule file will also let you define what level of help you want the agent to provide. The following are the supported help levels:

- **Low**: While you're building, the skill will only flag security risks. It won't suggest any improvements or alternatives to your architecture.
- **Medium**: While you're building, the skill might ask some clarifying questions if it detects potential issues. It won't suggest any improvements or alternatives to your architecture.
- **High**: While you're building, the skill will provide a high level of support. It will suggest improvements or alternatives to your architecture.

Work with multiple projects

Each project has its own set of credentials that AI coding tools can use to create and manage AWS resources on your behalf. You use profiles to access each project. A profile is a configuration setting and credential file maintained by the AWS CLI, and you give each profile a unique name.

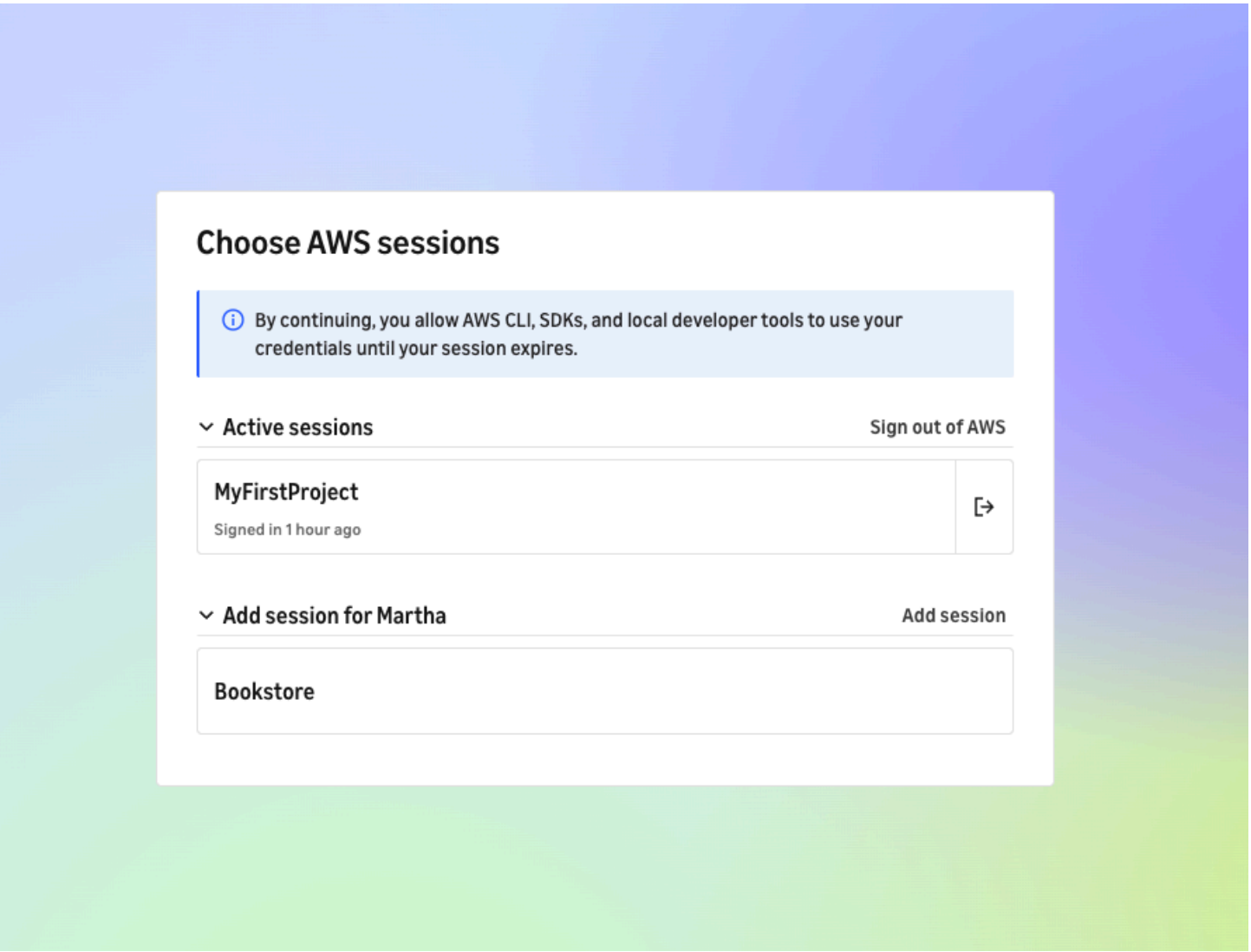
For example, if your first project was `MyFirstProject` and you chose the profile name `MyFirstProjectProfile` during setup, you can access that project by logging in to the AWS CLI with the following command:

```
aws login --profile MyFirstProjectProfile
```

If you create another project named `Bookstore`, you can access it by using the setup script and creating the profile `BookstoreProfile`. Then you can log in to the AWS CLI with the following command:

```
aws login --profile BookstoreProfile
```

However, when you run either of these commands, you'll need to choose a session for your AWS credentials on the **Choose AWS sessions** page.



This page shows the active sessions and sessions you can add. A session represents your authenticated state with your AWS credentials, and it provides the same credentials that your AI coding agent uses to create and manage resources.

In this example, you've already logged in to `MyFirstProject` and have not logged in to `Bookstore`. On the **Choose AWS sessions** page, you can do the following:

- To continue using AWS for `MyFirstProject`, choose `MyFirstProject`.
- To access `Bookstore`, choose `Bookstore`.
- To remove any of the current sessions, choose the exit icon.
- To access an AWS account using an IAM user, such as an account that you created with Sign up for AWS (advanced), choose **Add session**.

The credentials you choose will be the same credentials your AI coding agent uses to create and manage resources. For example, you might log in to the AWS CLI with the following command:

```
aws login --profile MyFirstProjectProfile
```

If you then choose the `Bookstore` session, your AI coding tool will create AWS resources in the `Bookstore` project.

Always choose the session that matches the project associated with your profile.

AWS actions for project owners

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you own your project, you have administrative control over your project. This includes tasks you can complete in Use AWS Settings. You also have the ability to perform root-only actions. These actions can't be completed by your team members.

The current root-only actions you can perform are the following:

- Edit or delete an S3 bucket policy that denies all principals. You might do this to unlock an S3 bucket with a misconfigured bucket policy.
- Edit or delete an SQS resource-based policy that denies all principals. You might do this to unlock an SQS queue with a misconfigured resource-based policy.

To perform a root-only action

1. Login into your project and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane of the console, choose **Root access management**.
3. For privilege action, choose the privileged action you want to take in the member account.
    - Select **Delete Amazon S3 bucket policy** to remove a misconfigured bucket policy that denies all principals from accessing the Amazon S3 bucket.
        a. Enter the S3 URI, or choose **Browse S3** to select a name from the buckets in your project.
        b. Choose **Delete bucket policy**.
        c. Use the Amazon S3 console to correct the bucket policy after deleting the misconfigured policy. For more information, see Adding a bucket policy by using the Amazon S3 console in the *Amazon S3 User Guide*.
    - Select **Delete Amazon SQS policy** to delete a resource-based policy that denies all principals from accessing an Amazon SQS queue.
        a. Enter the queue name in **SQS queue name**.
        b. Choose **Delete SQS policy**.
        c. Use the Amazon SQS console to correct the queue policy after deleting the misconfigured policy. For more information, see Configuring an access policy in Amazon SQS in the *Amazon SQS Developer Guide*.

Update your password in AWS Settings

Warning

We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can update your password in AWS Settings.

To update your password in AWS Settings

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage security and sign-in**.

   You'll be redirected to your AWS Builder ID profile.
3. For **Authorization**, choose **Update password**.
4. For **Current password**, enter your current password.
5. For **New password**, enter a new password.

   AWS requires that your password meet the following conditions:

   - It must have a minimum of 8 characters and a maximum of 128 characters.
   - It must include a minimum of three of the following mix of character types: uppercase, lowercase, numbers, and ! @ # $ % ^ & * () <> [] {} | _+-= symbols.
6. For **Confirm new password**, enter your new password.
7. Choose **Update**.

Manage multi-factor authentication (MFA) in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

Multi-factor authentication (MFA) is a simple and effective mechanism to enhance your security. The first factor — your password — is a secret that you memorize, also known as a knowledge factor. Other factors can be possession factors (something you have, such as a security key) or inherence factors (something you are, such as a biometric scan). We strongly recommend that you configure MFA to add an additional layer for your account.

You can register a built-in authenticator and also register a security key that you keep in a physically secure location. If you're unable to use your built-in authenticator, then you can use your registered security key. For authenticator applications, you can also enable the cloud backup or

sync feature in those apps. This helps you avoid losing access to your profile if you lose or break your MFA device.

Key points

- We recommend that you register multiple MFA devices. If you lose access to all registered MFA devices, you will be unable to recover your account.
- We recommend that you periodically review your registered MFA devices to ensure they are up to date and functional. Additionally, you should store those devices in a place that is physically secure when not in use.
- If you created your account using **Continue with Google**, you can enable multi-factor authentication through your Google account. For details, see Turn on 2-Step Verification.
- If you created your account using **Continue with Apple**, multi-factor authentication is likely already enabled in your Apple Account. If not, for details on how to enable it, see Two-factor authentication for Apple Account.
- If you created your account using **Continue with GitHub**, you can enable multi-factor authentication through your GitHub Account. For details, see Configuring (GitHub) two-factor authentication.
- If you created your account using **Continue with Amazon**, you can enable multi-factor authentication through your Amazon Account. For details, see What is Two-Step Verification?.

Available MFA types for our new AWS experience

Our new AWS experience supports the following multi-factor authentication (MFA) device types.

FIDO2 authenticators

FIDO2 is a standard that includes CTAP2 and WebAuthn and is based on public key cryptography. FIDO credentials are phishing-resistant because they are unique to the website that the credentials were created such as AWS.

AWS supports the two most common form factors for FIDO authenticators: built-in authenticators and security keys. See below for more information about the most common types of FIDO authenticators.

Built-in authenticators

Some devices have built-in authenticators, such as TouchID on MacBook or a Windows Hello-compatible camera. If your device is compatible with FIDO protocols, including WebAuthn, you can use your fingerprint or face as second factor. For more information, see FIDO Authentication.

Security keys

You can purchase a FIDO2-compatible external USB, BLE, or NFC-connected security key. When you're prompted for an MFA device, tap the key's sensor. YubiKey or Feitian make compatible devices. For a list of all compatible security keys, see FIDO Certified Products.

Password managers, passkey providers, and other FIDO authenticators

Multiple third party providers support FIDO authentication in mobile applications, as features in password managers, smart cards with a FIDO mode, and other form factors. These FIDO-compatible devices can work with IAM Identity Center, but we recommend that you test a FIDO authenticator yourself before enabling this option for MFA.

> **Note**
>
> Some FIDO authenticators can create discoverable FIDO credentials known as passkeys. Passkeys may be bound to the device that creates them, or they may be syncable and backed up to a cloud. For example, you can register a passkey using Apple Touch ID on a supported Macbook, and then log in to a site from a Windows laptop using Google Chrome with your passkey in iCloud by following the on-screen prompts at sign-in. For more information about which devices support syncable passkeys and current passkey interoperability between operating systems and browsers, see Device Support at passkeys.dev, a resource maintained by the FIDO Alliance And World Wide Web Consortium (W3C).

Authenticator applications

Authenticator apps are one-time password (OTP)-based third party-authenticators. You can use an authenticator application installed on your mobile device or tablet as an authorized MFA device. The third-party authenticator application must be compliant with RFC 6238, which is a standards-based time-based one-time password (TOTP) algorithm capable of generating six-digit authentication codes.

When prompted for MFA, you must enter a valid code from your authenticator app within the input box presented. Each MFA device assigned to a user must be unique. Two authenticator apps can be registered for any given user.

You can choose from the following well-known third-party authenticator apps. However, any TOTP-compliant application works with AWS Builder ID MFA.

Operating system	Tested authenticator app
Android	1Password, Authy, Duo Mobile, Microsoft Authenticator, Google Authenticator
iOS	1Password, Authy, Duo Mobile, Microsoft Authenticator, Google Authenticator

Register your MFA device

> **Note**
>
> After you sign up for MFA, sign out, and then sign in on the same device, you might not be prompted for MFA on trusted devices.

To register your MFA device using an authenticator app

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage security and sign-in**.

   You'll be redirected to your AWS Builder ID profile.
3. On the **Security** page, choose **Register device**.
4. On the **Register MFA device** page, choose **Authenticator app**.
5. AWS Builder ID operates and displays configuration information, including a QR code graphic. The graphic is a representation of the "secret configuration key" that is available for manual entry in authenticator apps that do not support QR codes.
6. Open your authenticator app. For a list of apps, see Authenticator applications.

If the authenticator app supports multiple MFA devices or accounts, choose the option to create a new MFA device or account.

7. Determine whether the MFA app supports QR codes, and then do one of the following on the **Set up your authenticator app** page:

   1. Choose **Show QR code**, and then use the app to scan the QR code. For example, you might choose the camera icon or choose an option similar to **Scan code**. Then use the device's camera to scan the code.
   2. Choose **Show secret key**, and then enter that secret key into your MFA app.

   When you finish, your authenticator app will generate and display a one-time password.

8. In the **Authenticator code** box, enter the one-time password that currently appears in your authenticator app. Choose **Assign MFA**.

   > **Important**
   >
   > Submit your request immediately after generating the code. If you generate the code and then wait too long to submit the request, the MFA device is successfully associated with your AWS Builder ID, but the MFA device is out of sync. This happens because time-based one-time passwords (TOTP) expire after a short period of time. If this happens, you can resync the device. For more information, see *I get the message 'An unexpected error has occurred' when I try to register or sign in with an authenticator app*.

9. To give your device a friendly name in AWS Builder ID, choose **Rename**. This name helps you distinguish this device from others that you register.

The MFA device is now ready for use with AWS Settings, your projects, and AWS Builder ID.

Register a security key as your MFA device

To register your MFA device using a security key

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage security and sign-in**.

You'll be redirected to your AWS Builder ID profile.

3. On the **Security** page, choose **Register device**.
4. On the **Register MFA device** page, choose **Security key**.
5. Ensure that your security key is enabled. If you use a separate physical security key, connect it to your computer.
6. Follow the instructions on your screen. Your experience varies based on your operating system and browser.
7. To give your device a friendly name in AWS Builder ID, choose **Rename**. This name helps you distinguish this device from others that you register.

The MFA device is now ready for use with AWS Settings, your projects, and AWS Builder ID.

Rename your MFA device

To rename your MFA device

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage security and sign-in**.

   You'll be redirected to your AWS Builder ID profile.
3. Choose **Security**. When you arrive at the page, you see that **Rename** is grayed out.
4. Select the MFA device that you want to change. This allows you to choose **Rename**. Then a dialog box appears.
5. In the prompt that opens, enter the new name in **MFA device name**, and choose **Rename**. The renamed device appears under **Multi-factor authentication (MFA) devices**.

Delete your MFA device

We recommend that you keep two or more active MFA devices. To disable multi-factor authentication, remove all registered MFA devices from your profile.

To delete an MFA device

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage security and sign-in**.

You'll be redirected to your AWS Builder ID profile.

3. On the **Security** page, select the MFA device that you want to change and choose **Delete**.
4. In the **Delete MFA device?** modal, follow the instructions to delete your device.
5. Choose **Delete**.

The deleted device no longer appears under **Multi-factor authentication (MFA) devices**.

Delete all sessions in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can delete all sessions across devices in AWS Settings. If there's a device you don't recognize, change your password and delete all sessions.

To delete all sessions

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage security and sign-in**.

   You'll be redirected to your AWS Builder ID profile.
3. For **Signed in devices**, choose **Delete all sessions**.
4. Confirm your choice and choose **Delete all sessions**.

You'll be logged out of AWS Settings.

Manage your name or nickname in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can change your name at any time. All team members can see your name, and if you've upgraded to the Paid Plan, this is the name that appears on your billing information. If you signed up for AWS in India, your name must exactly match the name that appears on the document you plan to use for customer verification.

When you change your name, you change the name in your AWS Builder ID. This will impact your name in AWS Settings, AWS Builder ID, and any AWS resources you access directly through the AWS Builder ID.

Your Name is how you're referred to in tools and services while interacting with others. Your Nickname indicates how you want to be known by AWS, friends, and other people you collaborate with closely.

To change your name or nickname

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**.
3. For **Profile info**, choose **Manage profile**.

   You'll be redirected to your AWS Builder ID profile.
4. For profile information, choose **Edit**.
5. (Optional) For **Name**, enter a new name.
6. (Optional) For **Nickname**, enter a new nickname.
7. Choose **Save changes**.

Change your email address in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can change your email in AWS Settings. When you change your email, your previous email will be separated from your AWS profile and AWS Builder ID. You won't be able to use that email to access any of the projects in AWS Settings, but you can use that email to sign up with AWS again.

Your email address is shared with collaborators when you give them access to your projects.

To change your email

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**.
3. For **Profile info**, choose **Manage profile**.

   You'll be redirected to your AWS Builder ID profile.
4. For **Contact information**, choose **Edit**.
5. For **Email address**, enter a new email address and choose **Verify email**. We will send you a verification code for your new email address. That will you enter in the **Verify email** popup box.
6. For **Verification code**, enter the verification code we sent to your new email.
7. Choose **Verify**.

Edit your contact information in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can edit your phone number or address associated with AWS Settings. When you modify these values, they might impact tax obligations.

To edit your contact information

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**.
3. For **Contact information**, choose **Edit**.
4. Enter new contact information.
5. Choose **Save**.

Opt out of use of your content for service improvement in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

AWS AI services may use and store customer content for service improvement, such as fixing operational issues, evaluating service performance, debugging, or model training. For this purpose, we might store such content in an AWS Region outside of the AWS Region where you are using the service. You can opt out of use of your content for service improvement through AWS Settings. The effect of opting out is that AWS attaches an AI services opt-out policy to your AWS organization. For more information about opt-out policies, see the AI services opt-out policies documentation.

Choosing to opt out from all services will opt all your projects out of service improvements using your content for current and future supported AI services supported by AI services opt-out policies.

Additional steps are required to opt out for Amazon Monitron. For more information, see AWS Service Terms. Only the project owner can opt out of data use by AWS AI services.

To opt out of data use by AWS AI services

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Project**.
3. For **Overview**, choose **Actions**, and then choose **Opt out of data use by AWS AI services**.
4. Confirm your choice and choose **Opt out**.

Request your AWS Builder ID data

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can request and view the personal information associated with your AWS Builder ID and the AWS applications and services you accessed with your AWS Builder ID. We will process your request within 30 days and notify you when your data is ready. While your data request is pending, you cannot close your AWS profile. You can still close projects or the organization that owns all your projects.

For more information about exercising your data subject rights, including for personal information provided in relation to other AWS websites, applications, products, services, events, and experiences, see AWS Privacy.

To request your data

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**.
3. Choose **Manage profile**.

   You'll be redirected to your AWS Builder ID profile.
4. Choose **Privacy & data**.
5. Choose to request your Builder ID data or your AWS data.

   A green confirmation message appears at the top of the page that we received your request and will complete it within 30 days.
6. When you receive an email from us that the request has been processed, navigate back to the **Privacy & data** page of your AWS Builder ID profile. Choose the newly available button **Download ZIP archive with your data**.

While your data request is pending, you will not be able to delete your AWS Builder ID.

Create a project in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

A project contains a single AWS account where you create AWS resources. A project also includes settings for sharing with other collaborators. The AWS resources you create should represent an application or a set of related use cases. One project can span the development and production environments of your application. For example, you can create a project called Bookstore that holds resources for a bookstore application including a database, compute resources, and storage.

You can share this project with other team members, and they can build and manage resources for the bookstore application.

Each project has its own billing based on the resources in the account. If you're using the paid plan, you can configure a spend limit in AWS Settings for each project. For more information, see Create a spend limit in AWS Settings. As the project owner, you are responsible for the monthly invoices for usage charges and recurring fees.

Considerations for creating a project

- You can access projects that you don't own as a team member.
- Each project name must be unique to your projects. Projects owned by other team members can have the same name.
- Project names can contain the following valid characters: a-z, A-Z, 0-9, and periods (.).
- When you create your project, AWS assigns it to a Region—US East (Ohio) (`us-east-2`), Europe (Stockholm) (`eu-north-1`), or Asia Pacific (Sydney) (`ap-southeast-2`)—based on the country in your contact address—either the United States, Europe, or Asia Pacific. Any Regional resources that you create will be hosted in that Region. For more information, see AWS Regions for your projects.

To create a project

To create a project

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Project**.
3. Choose **Create project**.
4. For **Project name**, enter a descriptive name for your project.

   This name is shared with team members when you invite them to collaborate.

5. Choose **Create**.

It takes a few seconds to create your project. When your project is available, you can choose the project and access the AWS Management Console to start building.

If you've already set up your AI coding tool to interface with your AWS account, you can prompt it to create resources in your new project.

Use the example prompts for your AI coding tool to get started:

```
What is the Region for my project test project?
```

```
Create a hello world Lambda function in my project test project.
```

```
Create an S3 bucket in my project test project.
```

AWS Regions for your projects

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you create your project, AWS assigns it to a Region—US East (Ohio) (`us-east-2`), Europe (Stockholm) (`eu-north-1`), or Asia Pacific (Sydney) (`ap-southeast-2`)—based on the country in your contact address—either the United States, Europe, or Asia Pacific. Any Regional resources that you create will be hosted in that Region.

The following are considerations for the selected AWS Region for your account:

- If you change your contact information in AWS Settings, your selected Region will not change. If you need to access a specific Region, use Sign up for AWS (advanced). You can also activate advanced features for your account to select different Regions.
- AWS may introduce additional supported Regions, but those will only be available for new sign-ups. To see the latest supported Regions as of August 18, 2026, see the tables below. If you already have a project, you can find your selected Region in AWS Settings.

- Global services are available with your project. These services are accessed via a single region that may differ from your home region. For more information, see AWS Fault Isolation Boundaries. In addition, there are some services which are available in specific Regions. In the AWS Management Console, you can view these services in certain Regions, but you'll return to your selected Region when you navigate to a different service.
- If you use Amazon Bedrock, you can use APIs that do not create resources in all commercial AWS Regions that are enabled by default.
- If you use CloudFront, certain features that would create resources outside of your selected Region aren't available. This includes using AWS WAF with CloudFront.
- Some CloudTrail events involving the global services are recorded in the Region where the global service operates. You have access to these events.
- AWS WAF and Amazon CloudWatch Logs are available in US East (N. Virginia) (`us-east-1`).

You can find what is the selected Region for all projects in your account by using AWS Settings. In the AWS Management Console, most service consoles are hosted in the Region you are in, including Bedrock and SageMaker. If you ever look at a service console from a global service, the console automatically routes all your API calls to the appropriate Region for the global service.

Find the selected AWS Region for your projects

As a best practice, we recommend that you always confirm the AWS Region for your project before you create any AWS resources.

To find the selected AWS Region for your projects

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Project**.
3. In the **Overview** section, choose **Additional Info**.

   In this view, you can also find the AWS account ID associated with your project.
4. For **Region**, you'll see the AWS Region your project has been provisioned in.

When you create Regional AWS resources in any of your projects, the Amazon Resource Names of these resources will include this AWS Region.

Countries where Regional resources are provisioned in US East (Ohio) (us-east-2)

Country name	Country code
Antigua and Barbuda	AG
Argentina	AR
Bahamas	BS
Barbados	BB
Belize	BZ
Bolivia	BO
Brazil	BR
Canada	CA
Chile	CL
Colombia	CO
Costa Rica	CR
Dominica	DM
Dominican Republic	DO
Ecuador	EC
El Salvador	SV
Grenada	GD
Guatemala	GT
Guyana	GY

Country name	Country code
Haiti	HT
Honduras	HN
Jamaica	JM
Mexico	MX
Nicaragua	NI
Panama	PA
Paraguay	PY
Peru	PE
Saint Kitts and Nevis	KN
Saint Lucia	LC
Saint Vincent and the Grenadines	VC
Suriname	SR
Trinidad and Tobago	TT
United States	US
Uruguay	UY
Venezuela	VE
Anguilla	AI
Aruba	AW
Bermuda	BM
Bonaire, Sint Eustatius and Saba	BQ

Country name	Country code
British Virgin Islands	VG
Cayman Islands	KY
Curaçao	CW
Falkland Islands	FK
French Guiana	GF
Guadeloupe	GP
Martinique	MQ
Montserrat	MS
Puerto Rico	PR
Saint Barthélemy	BL
Saint Martin	MF
Saint Pierre and Miquelon	PM
Sint Maarten	SX
Turks and Caicos Islands	TC
U.S. Virgin Islands	VI

Countries where Regional resources are provisioned in Asia Pacific (Sydney) (`ap-southeast-2`)

Country name	Country code
Afghanistan	AF

Country name	Country code
Australia	AU
Bangladesh	BD
Bhutan	BT
Brunei	BN
Cambodia	KH
China	CN
East Timor (Timor-Leste)	TL
Fiji	FJ
Hong Kong	HK
India	IN
Indonesia	ID
Japan	JP
Kazakhstan	KZ
Kiribati	KI
Kyrgyzstan	KG
Laos	LA
Macau	MO
Malaysia	MY
Maldives	MV
Marshall Islands	MH

Country name	Country code
Micronesia	FM
Mongolia	MN
Myanmar	MM
Nauru	NR
Nepal	NP
New Zealand	NZ
Pakistan	PK
Palau	PW
Papua New Guinea	PG
Philippines	PH
Samoa	WS
Singapore	SG
Solomon Islands	SB
South Korea	KR
Sri Lanka	LK
Taiwan	TW
Tajikistan	TJ
Thailand	TH
Tonga	TO
Turkmenistan	TM

Country name	Country code
Tuvalu	TV
Uzbekistan	UZ
Vanuatu	VU
Vietnam	VN
American Samoa	AS
Christmas Island	CX
Cocos (Keeling) Islands	CC
Cook Islands	CK
French Polynesia	PF
Guam	GU
Heard Island and McDonald Islands	HM
New Caledonia	NC
Niue	NU
Norfolk Island	NF
Northern Mariana Islands	MP
Pitcairn Islands	PN
Tokelau	TK
U.S. Minor Outlying Islands	UM
Wallis and Futuna	WF

Countries where Regional resources are provisioned in Europe (Stockholm) (`eu-north-1`)

Country name	Country code
Albania	AL
Algeria	DZ
Andorra	AD
Angola	AO
Armenia	AM
Austria	AT
Azerbaijan	AZ
Bahrain	BH
Belarus	BY
Belgium	BE
Benin	BJ
Bosnia and Herzegovina	BA
Botswana	BW
Bulgaria	BG
Burkina Faso	BF
Burundi	BI
Cabo Verde	CV
Cameroon	CM

Country name	Country code
Central African Republic	CF
Chad	TD
Comoros	KM
Congo (Brazzaville)	CG
Congo (DRC)	CD
Côte d'Ivoire	CI
Croatia	HR
Cyprus	CY
Czech Republic	CZ
Denmark	DK
Djibouti	DJ
Egypt	EG
Equatorial Guinea	GQ
Eritrea	ER
Estonia	EE
Eswatini	SZ
Ethiopia	ET
Finland	FI
France	FR
Gabon	GA

Country name	Country code
Gambia	GM
Georgia	GE
Germany	DE
Ghana	GH
Greece	GR
Guinea	GN
Guinea-Bissau	GW
Hungary	HU
Iceland	IS
Iraq	IQ
Ireland	IE
Israel	IL
Italy	IT
Jordan	JO
Kenya	KE
Kosovo	XK
Kuwait	KW
Latvia	LV
Lebanon	LB
Lesotho	LS

Country name	Country code
Liberia	LR
Libya	LY
Liechtenstein	LI
Lithuania	LT
Luxembourg	LU
Madagascar	MG
Malawi	MW
Mali	ML
Malta	MT
Mauritania	MR
Mauritius	MU
Moldova	MD
Monaco	MC
Montenegro	ME
Morocco	MA
Mozambique	MZ
Namibia	NA
Netherlands	NL
Niger	NE
Nigeria	NG

Country name	Country code
North Macedonia	MK
Norway	NO
Oman	OM
Palestine	PS
Poland	PL
Portugal	PT
Qatar	QA
Romania	RO
Russia	RU
Rwanda	RW
San Marino	SM
São Tomé and Príncipe	ST
Saudi Arabia	SA
Senegal	SN
Serbia	RS
Seychelles	SC
Sierra Leone	SL
Slovakia	SK
Slovenia	SI
Somalia	SO

Country name	Country code
South Africa	ZA
South Sudan	SS
Spain	ES
Sweden	SE
Switzerland	CH
Tanzania	TZ
Togo	TG
Tunisia	TN
Turkey	TR
Uganda	UG
Ukraine	UA
United Arab Emirates	AE
United Kingdom	GB
Vatican City	VA
Yemen	YE
Zambia	ZM
Zimbabwe	ZW
Åland Islands	AX
Antarctica	AQ
Bouvet Island	BV

Country name	Country code
British Indian Ocean Territory	IO
Faroe Islands	FO
French Southern Territories	TF
Gibraltar	GI
Greenland	GL
Guernsey	GG
Isle of Man	IM
Jersey	JE
Mayotte	YT
Réunion	RE
Saint Helena	SH
South Georgia and the South Sandwich Islands	GS
Svalbard and Jan Mayen	SJ
Western Sahara	EH

Managed policies for your organization

Warning

We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you use our new AWS experience, AWS manages the organization management policies including the resource control policies (RCPs) and the service control policies (SCPs). These

protective controls prevent a project owner or team member from performing actions that would inhibit the preconfigured defaults AWS has defined to help you build and develop quickly.

Service control policies for projects

The following is a service control policy for all users in a project and cannot be changed:

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "BlockOrgEscape",
      "Effect" : "Deny",
      "Action" : [
        "account:CloseAccount",
        "organizations:AcceptHandshake",
        "organizations:LeaveOrganization",
        "sso:CreateInstance"
      ],
      "Resource" : "*"
    },
    {
      "Sid" : "ProtectManagedRoles",
      "Effect" : "Deny",
      "Action" : [
        "iam:AttachRolePolicy",
        "iam:CreateRole",
        "iam:DeleteRole",
        "iam:DeleteRolePermissionsBoundary",
        "iam:DeleteRolePolicy",
        "iam:DetachRolePolicy",
        "iam:PutRolePermissionsBoundary",
        "iam:PutRolePolicy",
        "iam:TagRole",
        "iam:UntagRole",
        "iam:UpdateAssumeRolePolicy",
        "iam:UpdateRole"
      ],
      "Resource" : "arn:*:iam::*:role/managed/*",
      "Condition" : {
        "StringNotLike" : {
          "aws:PrincipalArn" : "arn:*:iam::*:role/managed/
AWSManagedAccountManagementAccessRole"
```

```
        }
      }
    }
  ]
}
```

This policy is not removed when you activate advanced features. For more information, see Activate advanced AWS features.

In addition, we apply the following service control policy for all users in a project to restrict certain modifications to your project that are either not supported or must be completed in AWS Settings:

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "BlockAccountManagementAPIs",
      "Effect" : "Deny",
      "Action" : [
        "account:AcceptPrimaryEmailUpdate",
        "account:DeleteAlternateContact",
        "account:DisableRegion",
        "account:EnableRegion",
        "account:GetAlternateContact",
        "account:GetContactInformation",
        "account:GetGovCloudAccountInformation",
        "account:GetRegionOptStatus",
        "account:PutAccountName",
        "account:PutAlternateContact",
        "account:PutContactInformation",
        "account:StartPrimaryEmailUpdate"
      ],
      "Resource" : "*"
    },
    {
      "Sid" : "DenyBillingOperations",
      "Effect" : "Deny",
      "Action" : "billing:*",
      "Resource" : "*"
    },
    {
      "Sid" : "DenyNotificationContactsOperations",
      "Effect" : "Deny",
```

```
      "Action" : "notifications-contacts:*",
      "Resource" : "*"
    },
    {
      "Sid" : "DenyRestrictedNotificationOperations",
      "Effect" : "Deny",
      "Action" : [
        "notifications:AssociateChannel",
        "notifications:AssociateManagedNotificationAccountContact",
        "notifications:AssociateManagedNotificationAdditionalChannel",
        "notifications:AssociateOrganizationalUnit",
        "notifications:CreateEventRule",
        "notifications:CreateNotificationConfiguration",
        "notifications:DeleteEventRule",
        "notifications:DeleteNotificationConfiguration",
        "notifications:DeregisterNotificationHub",
        "notifications:DisableNotificationsAccessForOrganization",
        "notifications:DisassociateChannel",
        "notifications:DisassociateManagedNotificationAccountContact",
        "notifications:DisassociateManagedNotificationAdditionalChannel",
        "notifications:DisassociateOrganizationalUnit",
        "notifications:EnableNotificationsAccessForOrganization",
        "notifications:GetEventRule",
        "notifications:GetFeatureOptInStatus",
        "notifications:GetNotificationConfiguration",
        "notifications:GetNotificationEvent",
        "notifications:GetNotificationsAccessForOrganization",
        "notifications:ListChannels",
        "notifications:ListEventRules",
        "notifications:ListManagedNotificationChannelAssociations",
        "notifications:ListMemberAccounts",
        "notifications:ListNotificationConfigurations",
        "notifications:ListNotificationEvents",
        "notifications:ListNotificationHubs",
        "notifications:ListOrganizationalUnits",
        "notifications:ListTagsForResource",
        "notifications:PutFeatureOptInStatus",
        "notifications:RegisterNotificationHub",
        "notifications:TagResource",
        "notifications:UntagResource",
        "notifications:UpdateEventRule",
        "notifications:UpdateNotificationConfiguration"
      ],
      "Resource" : "*"
```

```
    },
    {
      "Sid" : "DenyRoleManagerDisablement",
      "Effect" : "Deny",
      "Action" : "iam:PutAccountProperties",
      "Resource" : "*",
      "Condition" : {
        "ForAnyValue:StringEquals" : {
          "iam:AccountPropertyNamespaces" : "RoleManager"
        }
      }
    }
  ]
}
```

This policy is removed when you activate advanced features. For more information, see Activate advanced AWS features.

In addition, we also apply the following service control policy for all users in a project to restrict AWS Regions to allow for support for partitional services, depending on your *selected-region*:

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "UsEast1Partitional",
      "Effect" : "Deny",
      "NotAction" : [
        "account:*",
        "acm:*",
        "activate:*",
        "artifact:*",
        "aws-marketplace:*",
        "bedrock-mantle:CallWithBearerToken",
        "bedrock-mantle:CreateInference",
        "bedrock-mantle:GetInference",
        "bedrock-mantle:GetModel",
        "bedrock-mantle:ListModels",
        "bedrock:ApplyGuardrail",
        "bedrock:CountTokens",
        "bedrock:GetFoundationModel",
        "bedrock:GetFoundationModelAvailability",
        "bedrock:GetInferenceProfile",
```

```
        "bedrock:InvokeModel",
        "bedrock:InvokeModelWithResponseStream",
        "bedrock:ListFoundationModels",
        "bedrock:ListInferenceProfiles",
        "budgets:*",
        "ce:*",
        "chatbot:*",
        "cloudfront-keyvaluestore:*",
        "cloudfront:*",
        "cloudshell:*",
        "cloudtrail:LookupEvents",
        "cloudwatch:BatchGet*",
        "cloudwatch:Describe*",
        "cloudwatch:GenerateQuery",
        "cloudwatch:Get*",
        "cloudwatch:List*",
        "consoleapp:*",
        "cost-optimization-hub:*",
        "ec2:DescribeRegions",
        "ecr-public:*",
        "freetier:*",
        "health:*",
        "iam:*",
        "identitystore-auth:*",
        "identitystore:*",
        "invoicing:*",
        "kms:CreateGrant",
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:ListAliases",
        "lightsail:AttachCertificateToDistribution",
        "lightsail:CreateCertificate",
        "lightsail:CreateContactMethod",
        "lightsail:CreateDistribution",
        "lightsail:CreateDomain",
        "lightsail:CreateDomainEntry",
        "lightsail:DeleteAlarm",
        "lightsail:DeleteCertificate",
        "lightsail:DeleteContactMethod",
        "lightsail:DeleteDistribution",
        "lightsail:DeleteDomain",
        "lightsail:DeleteDomainEntry",
        "lightsail:DetachCertificateFromDistribution",
        "lightsail:Get*",
```

```
        "lightsail:IsVpcPeered",
        "lightsail:PutAlarm",
        "lightsail:ResetDistributionCache",
        "lightsail:SendContactMethodVerification",
        "lightsail:SetIpAddressType",
        "lightsail:TagResource",
        "lightsail:TestAlarm",
        "lightsail:UntagResource",
        "lightsail:UpdateDistribution",
        "lightsail:UpdateDistributionBundle",
        "lightsail:UpdateDomainEntry",
        "logs:*",
        "managedblockchain-query:*",
        "managedblockchain:*",
        "mapcredits:*",
        "notifications:GetManagedNotificationConfiguration",
        "notifications:GetManagedNotificationEvent",
        "notifications:ListManagedNotificationChildEvents",
        "notifications:ListManagedNotificationConfigurations",
        "notifications:ListManagedNotificationEvents",
        "organizations:*",
        "pricing:*",
        "q:*",
        "route53:*",
        "route53domains:*",
        "route53globalresolver:*",
        "s3:GetBucketLocation",
        "s3:ListAllMyBuckets",
        "servicequotas:*",
        "signin:*",
        "sso-directory:*",
        "sso-oauth:*",
        "sso:*",
        "sts:*",
        "support:*",
        "tax:*",
        "trustedadvisor:*",
        "uxc:*",
        "waf:*",
        "wafv2:*"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
```

```
          "aws:RequestedRegion" : "us-east-1"
        }
      }
    },
    {
      "Sid" : "DenyUsEast1WafResourceAssociation",
      "Effect" : "Deny",
      "Action" : [
        "wafv2:AssociateWebACL",
        "wafv2:DisassociateWebACL",
        "wafv2:GetWebACLForResource",
        "wafv2:ListResourcesForWebACL"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "aws:RequestedRegion" : "us-east-1"
        }
      }
    },
    {
      "Sid" : "UsWest2Partitional",
      "Effect" : "Deny",
      "NotAction" : [
        "bedrock-mantle:CallWithBearerToken",
        "bedrock-mantle:CreateInference",
        "bedrock-mantle:GetInference",
        "bedrock-mantle:GetModel",
        "bedrock-mantle:ListModels",
        "bedrock:ApplyGuardrail",
        "bedrock:CountTokens",
        "bedrock:GetFoundationModel",
        "bedrock:GetFoundationModelAvailability",
        "bedrock:GetInferenceProfile",
        "bedrock:InvokeModel",
        "bedrock:InvokeModelWithResponseStream",
        "bedrock:ListFoundationModels",
        "bedrock:ListInferenceProfiles",
        "cloudshell:*",
        "cloudtrail:LookupEvents",
        "health:*",
        "identitystore-auth:*",
        "identitystore:*",
        "kms:Decrypt",
```

```
        "kms:DescribeKey",
        "kms:ListAliases",
        "networkmanager-chat:*",
        "networkmanager:*",
        "route53-recovery-cluster:*",
        "route53-recovery-control-config:*",
        "route53-recovery-readiness:*",
        "servicequotas:*",
        "sso-directory:*",
        "sso-oauth:*",
        "sso:*",
        "support:*",
        "trustedadvisor:*",
        "uxc:*"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "aws:RequestedRegion" : "us-west-2"
        }
      }
    },
    {
      "Sid" : "RegionFloor",
      "Effect" : "Deny",
      "NotAction" : [
        "bedrock-mantle:CallWithBearerToken",
        "bedrock-mantle:CreateInference",
        "bedrock-mantle:GetInference",
        "bedrock-mantle:GetModel",
        "bedrock-mantle:ListModels",
        "bedrock:ApplyGuardrail",
        "bedrock:CountTokens",
        "bedrock:GetFoundationModel",
        "bedrock:GetFoundationModelAvailability",
        "bedrock:GetInferenceProfile",
        "bedrock:InvokeModel",
        "bedrock:InvokeModelWithResponseStream",
        "bedrock:ListFoundationModels",
        "bedrock:ListInferenceProfiles"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringNotEquals" : {
```

```
          "aws:RequestedRegion" : [
            "unspecified",
            "us-east-1",
            "selected-region",
            "us-west-2"
          ]
        }
      }
    }
  ]
}
```

After you activate advanced features, the following service control policy is applied to your account:

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "UsEast1Partitional",
      "Effect" : "Deny",
      "NotAction" : [
        "a4b:*",
        "access-analyzer:*",
        "account-access:*",
        "account:*",
        "acm:*",
        "activate:*",
        "artifact:*",
        "aws-marketplace-management:*",
        "aws-marketplace:*",
        "aws-portal:*",
        "bedrock-mantle:CallWithBearerToken",
        "bedrock-mantle:CreateInference",
        "bedrock-mantle:GetInference",
        "bedrock-mantle:GetModel",
        "bedrock-mantle:ListModels",
        "bedrock:ApplyGuardrail",
        "bedrock:CountTokens",
        "bedrock:GetFoundationModel",
        "bedrock:GetFoundationModelAvailability",
        "bedrock:GetInferenceProfile",
        "bedrock:InvokeModel",
```

```
        "bedrock:InvokeModelWithResponseStream",
        "bedrock:ListFoundationModels",
        "bedrock:ListInferenceProfiles",
        "billing:*",
        "billingconductor:*",
        "budgets:*",
        "builderid:*",
        "ce:*",
        "chatbot:*",
        "chime:*",
        "cloudfront-keyvaluestore:*",
        "cloudfront:*",
        "cloudshell:*",
        "cloudtrail:LookupEvents",
        "cloudwatch:BatchGet*",
        "cloudwatch:Describe*",
        "cloudwatch:GenerateQuery",
        "cloudwatch:Get*",
        "cloudwatch:List*",
        "compute-optimizer:*",
        "config:*",
        "consoleapp:*",
        "consolidatedbilling:*",
        "cost-optimization-hub:*",
        "cur:*",
        "datapipeline:GetAccountLimits",
        "devicefarm:*",
        "directconnect:*",
        "ec2:DescribeRegions",
        "ec2:DescribeTransitGateways",
        "ec2:DescribeVpnGateways",
        "ecr-public:*",
        "fms:*",
        "freetier:*",
        "globalaccelerator:*",
        "health:*",
        "iam:*",
        "identitystore-auth:*",
        "identitystore:*",
        "importexport:*",
        "invoicing:*",
        "iq:*",
        "kms:*",
        "license-manager:ListReceivedLicenses",
```

```
        "lightsail:AttachCertificateToDistribution",
        "lightsail:CreateCertificate",
        "lightsail:CreateContactMethod",
        "lightsail:CreateDistribution",
        "lightsail:CreateDomain",
        "lightsail:CreateDomainEntry",
        "lightsail:DeleteAlarm",
        "lightsail:DeleteCertificate",
        "lightsail:DeleteContactMethod",
        "lightsail:DeleteDistribution",
        "lightsail:DeleteDomain",
        "lightsail:DeleteDomainEntry",
        "lightsail:DetachCertificateFromDistribution",
        "lightsail:Get*",
        "lightsail:IsVpcPeered",
        "lightsail:PutAlarm",
        "lightsail:ResetDistributionCache",
        "lightsail:SendContactMethodVerification",
        "lightsail:SetIpAddressType",
        "lightsail:TagResource",
        "lightsail:TestAlarm",
        "lightsail:UntagResource",
        "lightsail:UpdateDistribution",
        "lightsail:UpdateDistributionBundle",
        "lightsail:UpdateDomainEntry",
        "logs:*",
        "managedblockchain-query:*",
        "managedblockchain:*",
        "mapcredits:*",
        "mobileanalytics:*",
        "networkmanager:*",
        "notifications-contacts:*",
        "notifications:*",
        "organizations:*",
        "payments:*",
        "pricing:*",
        "q:*",
        "quicksight:DescribeAccountSubscription",
        "resource-explorer-2:*",
        "route53-recovery-cluster:*",
        "route53-recovery-control-config:*",
        "route53-recovery-readiness:*",
        "route53:*",
        "route53domains:*",
```

```
        "route53globalresolver:*",
        "s3:CreateMultiRegionAccessPoint",
        "s3:DeleteMultiRegionAccessPoint",
        "s3:DescribeMultiRegionAccessPointOperation",
        "s3:GetAccountPublicAccessBlock",
        "s3:GetBucketLocation",
        "s3:GetBucketPolicyStatus",
        "s3:GetBucketPublicAccessBlock",
        "s3:GetMultiRegionAccessPoint",
        "s3:GetMultiRegionAccessPointPolicy",
        "s3:GetMultiRegionAccessPointPolicyStatus",
        "s3:GetStorageLensConfiguration",
        "s3:GetStorageLensDashboard",
        "s3:ListAllMyBuckets",
        "s3:ListMultiRegionAccessPoints",
        "s3:ListStorageLensConfigurations",
        "s3:PutAccountPublicAccessBlock",
        "s3:PutMultiRegionAccessPointPolicy",
        "savingsplans:*",
        "servicequotas:*",
        "shield:*",
        "signin:*",
        "sso-directory:*",
        "sso-oauth:*",
        "sso:*",
        "sts:*",
        "support:*",
        "supportapp:*",
        "supportplans:*",
        "sustainability:*",
        "tag:GetResources",
        "tax:*",
        "trustedadvisor:*",
        "uxc:*",
        "vendor-insights:ListEntitledSecurityProfiles",
        "waf-regional:*",
        "waf:*",
        "wafv2:*"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
          "aws:RequestedRegion" : "us-east-1"
        }
```

```
      }
    },
    {
      "Sid" : "UsWest2Partitional",
      "Effect" : "Deny",
      "NotAction" : [
        "bedrock-mantle:CallWithBearerToken",
        "bedrock-mantle:CreateInference",
        "bedrock-mantle:GetInference",
        "bedrock-mantle:GetModel",
        "bedrock-mantle:ListModels",
        "bedrock:ApplyGuardrail",
        "bedrock:CountTokens",
        "bedrock:GetFoundationModel",
        "bedrock:GetFoundationModelAvailability",
        "bedrock:GetInferenceProfile",
        "bedrock:InvokeModel",
        "bedrock:InvokeModelWithResponseStream",
        "bedrock:ListFoundationModels",
        "bedrock:ListInferenceProfiles",
        "cloudshell:*",
        "cloudtrail:LookupEvents",
        "health:*",
        "identitystore-auth:*",
        "identitystore:*",
        "kms:Decrypt",
        "kms:DescribeKey",
        "kms:ListAliases",
        "networkmanager-chat:*",
        "networkmanager:*",
        "route53-recovery-cluster:*",
        "route53-recovery-control-config:*",
        "route53-recovery-readiness:*",
        "servicequotas:*",
        "sso-directory:*",
        "sso-oauth:*",
        "sso:*",
        "support:*",
        "trustedadvisor:*",
        "uxc:*"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringEquals" : {
```

```
          "aws:RequestedRegion" : "us-west-2"
        }
      }
    },
    {
      "Sid" : "RegionFloor",
      "Effect" : "Deny",
      "NotAction" : [
        "bedrock-mantle:CallWithBearerToken",
        "bedrock-mantle:CreateInference",
        "bedrock-mantle:GetInference",
        "bedrock-mantle:GetModel",
        "bedrock-mantle:ListModels",
        "bedrock:ApplyGuardrail",
        "bedrock:CountTokens",
        "bedrock:GetFoundationModel",
        "bedrock:GetFoundationModelAvailability",
        "bedrock:GetInferenceProfile",
        "bedrock:InvokeModel",
        "bedrock:InvokeModelWithResponseStream",
        "bedrock:ListFoundationModels",
        "bedrock:ListInferenceProfiles"
      ],
      "Resource" : "*",
      "Condition" : {
        "StringNotEquals" : {
          "aws:RequestedRegion" : [
            "unspecified",
            "us-east-1",
            "selected-region",
            "us-west-2"
          ]
        }
      }
    }
  ]
}
```

Service control policies for spend limits

If you use a spend limit, the following service control policies are applied to your project as it nears the spend limit. For more information about spend limits, see Create a spend limit in AWS Settings.

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "DenyBedrockUsage",
    "Effect": "Deny",
    "Action": [
      "bedrock:ApplyGuardrail",
      "bedrock:CallWithBearerToken",
      "bedrock:InvokeAgent",
      "bedrock:InvokeAutomatedReasoningPolicy",
      "bedrock:InvokeDataAutomation",
      "bedrock:InvokeDataAutomationAsync",
      "bedrock:InvokeFlow",
      "bedrock:InvokeInlineAgent",
      "bedrock:InvokeModel",
      "bedrock:InvokeModelWithResponseStream",
      "bedrock:InvokeTool",
      "bedrock:OptimizePrompt",
      "bedrock:Retrieve",
      "bedrock:RetrieveAndGenerate",
      "bedrock:StartFlowExecution",
      "bedrock-mantle:CallWithBearerToken",
      "bedrock-mantle:CreateInference"
    ],
    "Resource": "*"
  }
}
```

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "DenyLambdaUsage",
    "Effect": "Deny",
    "Action": [
      "lambda:DeleteFunctionConcurrency",
      "lambda:InvokeAsync",
      "lambda:InvokeFunction",
      "lambda:InvokeFunctionUrl",
      "lambda:PutFunctionConcurrency"
    ],
    "Resource": "*"
  }
```

```
}
```

```
{
  "Version": "2012-10-17",
  "Statement": {
    "Sid": "DenySageMakerInference",
    "Effect": "Deny",
    "Action": [
      "sagemaker:InvokeEndpoint",
      "sagemaker:InvokeEndpointAsync",
      "sagemaker:InvokeEndpointWithResponseStream"
    ],
    "Resource": "*"
  }
}
```

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "DenyCompute",
      "Effect": "Deny",
      "Action": [
        "ec2:CreateNatGateway",
        "ec2:CreateVolume",
        "ec2:CreateVpnConnection",
        "ec2:RunInstances",
        "ec2:StartInstances",
        "elasticloadbalancing:CreateLoadBalancer",
        "autoscaling:CreateAutoScalingGroup",
        "ecs:CreateService",
        "eks:CreateCluster",
        "eks:CreateNodegroup",
        "eks:CreateFargateProfile",
        "elasticbeanstalk:CreateEnvironment",
        "apprunner:CreateService",
        "apprunner:ResumeService",
        "batch:CreateComputeEnvironment",
        "elasticmapreduce:RunJobFlow",
        "appstream:CreateFleet"
      ],
      "Resource": "*"
    },
```

```
    {
      "Sid": "DenyLambda",
      "Effect": "Deny",
      "Action": "lambda:CreateFunction",
      "Resource": "*"
    },
    {
      "Sid": "DenyDatabase",
      "Effect": "Deny",
      "Action": [
        "rds:CreateDBCluster",
        "rds:CreateDBInstance",
        "rds:RestoreDBClusterFromS3",
        "rds:RestoreDBClusterFromSnapshot",
        "rds:RestoreDBClusterToPointInTime",
        "rds:RestoreDBInstanceFromDBSnapshot",
        "rds:RestoreDBInstanceFromS3",
        "rds:RestoreDBInstanceToPointInTime",
        "rds:StartDBCluster",
        "rds:StartDBInstance",
        "dynamodb:CreateTable",
        "dynamodb:CreateGlobalTable",
        "elasticache:CreateCacheCluster",
        "elasticache:CreateReplicationGroup",
        "timestream:CreateDatabase",
        "timestream:CreateTable",
        "memorydb:CreateCluster",
        "es:CreateDomain",
        "aoss:CreateCollection",
        "redshift:CreateCluster",
        "redshift:ResumeCluster",
        "redshift-serverless:CreateWorkgroup",
        "dsql:CreateCluster"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyStorage",
      "Effect": "Deny",
      "Action": [
        "elasticfilesystem:CreateFileSystem",
        "datasync:CreateTask",
        "storagegateway:ActivateGateway",
        "glacier:CreateVault"
```

```
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenySageMaker",
      "Effect": "Deny",
      "Action": [
        "sagemaker:CreateApp",
        "sagemaker:CreateEndpoint",
        "sagemaker:CreateNotebookInstance",
        "sagemaker:CreateProcessingJob",
        "sagemaker:CreateTrainingJob",
        "sagemaker:CreateTransformJob",
        "sagemaker:StartNotebookInstance"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyBedrock",
      "Effect": "Deny",
      "Action": [
        "bedrock:CreateCustomModelDeployment",
        "bedrock:CreateDataAutomationProject",
        "bedrock:CreateEvaluationJob",
        "bedrock:CreateMarketplaceModelEndpoint",
        "bedrock:CreateModelCustomizationJob",
        "bedrock:CreateModelImportJob",
        "bedrock:CreateModelInvocationJob",
        "bedrock:CreateProvisionedModelThroughput",
        "bedrock-mantle:CreateCustomizedModel",
        "bedrock-mantle:CreateFineTuningJob",
        "bedrock-mantle:CreateReservation"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyAIML",
      "Effect": "Deny",
      "Action": [
        "kendra:CreateIndex",
        "personalize:CreateSolution",
        "personalize:CreateCampaign",
        "forecast:CreatePredictor",
        "forecast:CreateForecast",
```

```
        "rekognition:CreateProject",
        "lex:CreateBot"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyNetworking",
      "Effect": "Deny",
      "Action": [
        "cloudfront:CreateDistribution",
        "route53:CreateHostedZone",
        "appmesh:CreateMesh"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyApiGatewayCreate",
      "Effect": "Deny",
      "Action": "apigateway:POST",
      "Resource": [
        "arn:aws:apigateway:*::/restapis",
        "arn:aws:apigateway:*::/apis"
      ]
    },
    {
      "Sid": "DenyMessaging",
      "Effect": "Deny",
      "Action": [
        "mq:CreateBroker",
        "kafka:CreateCluster",
        "kafka:CreateVpcConnection",
        "kinesis:CreateStream",
        "firehose:CreateDeliveryStream",
        "kinesisanalytics:CreateApplication",
        "kinesisvideo:CreateStream",
        "sns:CreateTopic",
        "sqs:CreateQueue",
        "events:CreateEventBus",
        "pipes:CreatePipe",
        "scheduler:CreateSchedule"
      ],
      "Resource": "*"
    },
    {
```

```
      "Sid": "DenyAnalytics",
      "Effect": "Deny",
      "Action": [
        "states:CreateStateMachine",
        "glue:CreateJob",
        "glue:CreateCrawler",
        "athena:CreateWorkGroup",
        "appsync:CreateGraphqlApi",
        "appflow:CreateFlow",
        "datazone:CreateDomain",
        "datazone:CreateProject",
        "entityresolution:CreateMatchingWorkflow",
        "datapipeline:CreatePipeline",
        "dataexchange:CreateDataSet"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyDevTools",
      "Effect": "Deny",
      "Action": [
        "amplify:CreateApp",
        "amplify:CreateBranch",
        "cloud9:CreateEnvironmentEC2",
        "codebuild:CreateProject",
        "codepipeline:CreatePipeline",
        "codedeploy:CreateApplication",
        "cloudformation:CreateStack",
        "cloudformation:CreateStackSet",
        "codeartifact:CreateRepository",
        "codeartifact:CreateDomain"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenySecurity",
      "Effect": "Deny",
      "Action": [
        "kms:CreateKey",
        "wafv2:CreateWebACL",
        "cloudhsm:CreateCluster",
        "network-firewall:CreateFirewall",
        "secretsmanager:CreateSecret",
        "acm-pca:CreateCertificateAuthority",
```

```
        "cognito-idp:CreateUserPool",
        "cognito-identity:CreateIdentityPool",
        "ds:CreateDirectory",
        "ds:CreateMicrosoftAD",
        "fms:PutPolicy",
        "payment-cryptography:CreateKey"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyMedia",
      "Effect": "Deny",
      "Action": [
        "medialive:CreateChannel",
        "mediapackage:CreateChannel",
        "mediaconnect:CreateFlow",
        "mediatailor:CreateChannel"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyIoT",
      "Effect": "Deny",
      "Action": [
        "iot:CreateThing",
        "iot:CreateTopicRule",
        "iotevents:CreateDetectorModel",
        "iottwinmaker:CreateWorkspace",
        "iotsitewise:CreateAssetModel",
        "iotsitewise:CreatePortal",
        "iotfleetwise:CreateCampaign"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyMisc",
      "Effect": "Deny",
      "Action": [
        "ses:CreateConfigurationSet",
        "ecr:CreateRepository",
        "aps:CreateWorkspace",
        "healthlake:CreateFHIRDatastore",
        "transfer:CreateServer",
        "managedblockchain:CreateNetwork",
```

```
        "managedblockchain:CreateMember",
        "route53-recovery-control-config:CreateCluster",
        "fis:CreateExperimentTemplate",
        "appfabric:CreateAppBundle",
        "appfabric:CreateIngestion",
        "b2bi:CreateProfile",
        "b2bi:CreateTransformer",
        "geo:CreateMap",
        "geo:CreatePlaceIndex",
        "geo:CreateTracker",
        "geo:CreateGeofenceCollection",
        "geo:CreateRouteCalculator"
      ],
      "Resource": "*"
    }
  ]
}
```

Service control policies for the Free Tier for projects

The following service control policy controls access to the AWS services available in the Free Tier of our new AWS experience. This policy cannot be modified. To access additional services, upgrade your account. For more information, see Upgrade your account in AWS Settings.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowFreeTierServices",
      "Effect": "Deny",
      "NotAction": [
        "a2c:*",
        "access-analyzer:CheckAccessNotGranted",
        "access-analyzer:CheckNoNewAccess",
        "access-analyzer:CheckNoPublicAccess",
        "access-analyzer:ValidatePolicy",
        "account:*",
        "acm:*",
        "aco-automation:*",
        "action-recommendations:*",
        "activate:*",
        "aiops:*",
        "airflow-serverless:*",
```

```
        "amplify:*",
        "amplifybackend:*",
        "amplifyuibuilder:*",
        "aoss:*",
        "apigateway:*",
        "app-integrations:*",
        "appconfig:*",
        "appflow:*",
        "application-autoscaling:*",
        "application-signals-mcp:*",
        "application-signals:*",
        "application-transformation:*",
        "applicationinsights:*",
        "appmesh:*",
        "appsync:*",
        "aps:*",
        "arc-zonal-shift:*",
        "artifact:*",
        "athena:*",
        "autoscaling-plans:*",
        "autoscaling:*",
        "aws-external-anthropic:*",
        "aws-marketplace:*",
        "b2bi:*",
        "backup-gateway:*",
        "backup-search:*",
        "backup-storage:*",
        "backup:*",
        "batch:*",
        "bcm-pricing-calculator:*",
        "bedrock-agentcore:*",
        "bedrock-mantle:*",
        "bedrock-websearch:*",
        "bedrock:*",
        "budgets:*",
        "ce:*",
        "chatbot:*",
        "cloud9:*",
        "cloudformation:*",
        "cloudfront-keyvaluestore:*",
        "cloudfront:*",
        "cloudshell:*",
        "cloudtrail-data:*",
        "cloudtrail:*",
```

```
        "cloudwatch:*",
        "codeartifact:*",
        "codebuild:*",
        "codecommit:*",
        "codeconnections:*",
        "codedeploy-commands-secure:*",
        "codedeploy:*",
        "codepipeline:*",
        "codestar-connections:*",
        "codestar-notifications:*",
        "cognito-identity:*",
        "cognito-idp:*",
        "cognito-sync:*",
        "compute-optimizer:*",
        "config:*",
        "consoleapp:*",
        "consolidatedbilling:*",
        "cost-optimization-hub:*",
        "customer-verification:*",
        "databrew:*",
        "datapipeline:*",
        "datasync:*",
        "datazone:*",
        "dbqms:*",
        "dlm:*",
        "ds-data:*",
        "ds:*",
        "dsql:*",
        "dynamodb:*",
        "ebs:*",
        "ec2-instance-connect:*",
        "ec2:*",
        "ec2messages:*",
        "ecr-public:*",
        "ecr:*",
        "ecs-mcp:*",
        "ecs:*",
        "eks-auth:*",
        "eks-mcp:*",
        "eks:*",
        "elasticache:*",
        "elasticbeanstalk:*",
        "elasticfilesystem:*",
        "elasticloadbalancing:*",
```

```
        "entityresolution:*",
        "es:*",
        "events:*",
        "evidently:*",
        "execute-api:*",
        "finops-agent:*",
        "freertos:*",
        "freetier:*",
        "geo-maps:*",
        "geo-places:*",
        "geo-routes:*",
        "geo:*",
        "glue:*",
        "groundtruthlabeling:*",
        "health:*",
        "iam:*",
        "identitystore-auth:*",
        "identitystore:*",
        "imagebuilder:*",
        "internetmonitor:*",
        "invoicing:*",
        "iot-device-tester:*",
        "iot:*",
        "iotdeviceadvisor:*",
        "iotfleethub:*",
        "iotjobsdata:*",
        "iotmanagedintegrations:*",
        "iotsitewise:*",
        "iottwinmaker:*",
        "iotwireless:*",
        "kms:*",
        "lakeformation:*",
        "lambda:*",
        "launchwizard:*",
        "lex:*",
        "license-manager-linux-subscriptions:*",
        "license-manager-user-subscriptions:*",
        "license-manager:*",
        "logs:*",
        "mgn:*",
        "migrationhub-orchestrator:*",
        "migrationhub-strategy:*",
        "mpa:*",
        "mq:*",
```

```
        "networkflowmonitor:*",
        "networkmanager-chat:*",
        "networkmanager:*",
        "networkmonitor:*",
        "notifications-contacts:*",
        "notifications:*",
        "nova-act:*",
        "oam:*",
        "observabilityadmin:*",
        "opensearch:*",
        "organizations:*",
        "osis:*",
        "payment-cryptography:*",
        "pi:*",
        "pipes:*",
        "polly:*",
        "pricing:*",
        "q:DeleteConversation",
        "q:GenerateCodeFromCommands",
        "q:GenerateCodeRecommendations",
        "q:GetConversation",
        "q:GetIdentityMetadata",
        "q:GetTroubleshootingResults",
        "q:ListConversations",
        "q:PassRequest",
        "q:SendEvent",
        "q:SendMessage",
        "q:StartConversation",
        "q:StartTroubleshootingAnalysis",
        "q:StartTroubleshootingResolutionExplanation",
        "q:UpdateConversation",
        "q:UpdateTroubleshootingCommandResult",
        "q:UsePlugin",
        "ram:*",
        "rbin:*",
        "rds-data:*",
        "rds-db:*",
        "rds:*",
        "rekognition:*",
        "resource-explorer-2:*",
        "resource-explorer:*",
        "resource-groups:*",
        "rhelkb:*",
        "route53-recovery-cluster:*",
```

```
        "route53-recovery-control-config:*",
        "route53-recovery-readiness:*",
        "route53:*",
        "route53domains:*",
        "route53globalresolver:*",
        "route53profiles:*",
        "route53resolver:*",
        "rum:*",
        "s3-object-lambda:*",
        "s3:*",
        "s3express:*",
        "s3files:*",
        "s3tables:*",
        "s3vectors:*",
        "sagemaker-data-science-assistant:*",
        "sagemaker-geospatial:*",
        "sagemaker-mlflow:*",
        "sagemaker-unified-studio-mcp:*",
        "sagemaker:*",
        "scheduler:*",
        "schemas:*",
        "secretsmanager:*",
        "serverlessrepo:*",
        "servicecatalog:Describe*",
        "servicecatalog:List*",
        "servicecatalog:ProvisionProduct",
        "servicecatalog:ScanProvisionedProducts",
        "servicecatalog:SearchProducts",
        "servicecatalog:SearchProvisionedProducts",
        "servicecatalog:TerminateProvisionedProduct",
        "servicecatalog:UpdateProvisionedProduct",
        "servicediscovery:*",
        "serviceextract:*",
        "servicequotas:*",
        "ses:*",
        "signer:*",
        "signin:*",
        "sns:*",
        "sqlworkbench:*",
        "sqs:*",
        "ssm-contacts:*",
        "ssm-guiconnect:*",
        "ssm-incidents:*",
        "ssm-quicksetup:*",
```

```
        "ssm-sap:*",
        "ssm:*",
        "ssmmessages:*",
        "sso-directory:*",
        "sso-oauth:*",
        "sso:*",
        "states:*",
        "sts:*",
        "support:*",
        "swf:*",
        "synthetics:*",
        "tag:*",
        "tax:*",
        "timestream-influxdb:*",
        "ts:*",
        "uxc:*",
        "verifiedpermissions:*",
        "vpce:*",
        "waf-regional:*",
        "waf:*",
        "wafv2:*",
        "wellarchitected:*",
        "xray:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyEC2CapacityBlock",
      "Effect": "Deny",
      "Action": ["ec2:PurchaseCapacityBlock", "ec2:PurchaseCapacityBlockExtension"],
      "Resource": "*"
    },
    {
      "Sid": "DenyRestrictedAgreements",
      "Effect": "Deny",
      "Action": ["artifact:AcceptAgreement", "artifact:AcceptNdaForAgreement"],
      "Resource": [
        "arn:aws:artifact:::agreement/agreement-9c1kBcYznTkcpRIm",
        "arn:aws:artifact:::agreement/agreement-y03aUwMAEorHtqjv",
        "arn:aws:artifact:::agreement/agreement-bexgr7sjvXAW4Gxu",
        "arn:aws:artifact:::agreement/agreement-HZTdNwJuqOKLReXC"
      ]
    },
    {
```

```
      "Sid": "DenyDocDB",
      "Effect": "Deny",
      "Action": ["rds:CreateDBCluster", "rds:CreateDBInstance"],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "rds:DatabaseEngine": ["docdb"]
        }
      }
    },
    {
      "Sid": "DenyCloudWAN",
      "Effect": "Deny",
      "Action": [
        "networkmanager:CreateCoreNetwork",
        "networkmanager:CreateConnectAttachment",
        "networkmanager:CreateConnectPeer",
        "networkmanager:CreateVpcAttachment",
        "networkmanager:CreateSiteToSiteVpnAttachment",
        "networkmanager:CreateTransitGatewayPeering",
        "networkmanager:CreateDirectConnectGatewayAttachment",
        "networkmanager:PutCoreNetworkPolicy",
        "networkmanager:ExecuteCoreNetworkChangeSet"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyTransitGatewayAndVPN",
      "Effect": "Deny",
      "Action": [
        "ec2:CreateTransitGateway",
        "ec2:CreateTransitGatewayVpcAttachment",
        "ec2:CreateTransitGatewayConnect",
        "ec2:CreateTransitGatewayConnectPeer",
        "ec2:CreateTransitGatewayMulticastDomain",
        "ec2:CreateTransitGatewayPeeringAttachment",
        "ec2:CreateTransitGatewayPolicyTable",
        "ec2:CreateTransitGatewayPrefixListReference",
        "ec2:CreateTransitGatewayRoute",
        "ec2:CreateTransitGatewayRouteTable",
        "ec2:CreateTransitGatewayRouteTableAnnouncement",
        "ec2:CreateClientVpnEndpoint",
        "ec2:CreateClientVpnRoute",
        "ec2:AuthorizeClientVpnIngress",
```

```
        "ec2:AssociateClientVpnTargetNetwork",
        "ec2:CreateVpnConnection",
        "ec2:CreateVpnConnectionRoute",
        "ec2:CreateVpnGateway",
        "ec2:CreateCustomerGateway"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenySageMakerGroundTruth",
      "Effect": "Deny",
      "Action": [
        "sagemaker:CreateLabelingJob",
        "sagemaker:CreateWorkteam",
        "sagemaker:CreateWorkforce",
        "sagemaker:CreateHumanTaskUi"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyGlacierStorageClasses",
      "Effect": "Deny",
      "Action": ["s3:PutObject"],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "s3:x-amz-storage-class": ["DEEP_ARCHIVE", "GLACIER"]
        }
      }
    },
    {
      "Sid": "DenyEC2Fleet",
      "Effect": "Deny",
      "Action": ["ec2:CreateFleet"],
      "Resource": "*"
    },
    {
      "Sid": "DenyEC2BYOLAndLicenseConfigurations",
      "Effect": "Deny",
      "Action": ["ec2:RunInstances", "ec2:StartInstances"],
      "Resource": "arn:aws:ec2:*:*:instance/*",
      "Condition": {
        "StringEquals": {
          "ec2:Tenancy": "host"
```

```
        }
      }
    },
    {
      "Sid": "DenyEC2InstanceStoreRootDevice",
      "Effect": "Deny",
      "Action": ["ec2:RunInstances"],
      "Resource": "arn:aws:ec2:*:*:instance/*",
      "Condition": {
        "StringEquals": {
          "ec2:RootDeviceType": "instance-store"
        }
      }
    },
    {
      "Sid": "DenyLicenseManagerAssociation",
      "Effect": "Deny",
      "Action": ["license-manager:CreateLicenseConfiguration"],
      "Resource": "*"
    },
    {
      "Sid": "DenyIAMRestrictedActions",
      "Effect": "Deny",
      "Action": [
        "iam:*Alias*",
        "iam:*Organizations*",
        "iam:*Provider*",
        "iam:SetSecurityTokenServicePreferences",
        "iam:*LoginProfile*",
        "iam:CreateGroup",
        "iam:AttachGroupPolicy",
        "iam:AddUserToGroup",
        "iam:PutGroupPolicy",
        "iam:UpdateGroup",
        "iam:UpdateServerCertificate",
        "iam:UpdateSigningCertificate",
        "iam:UploadServerCertificate",
        "iam:UploadSigningCertificate",
        "iam:TagServerCertificate",
        "iam:UntagServerCertificate"
      ],
      "Resource": "*"
    },
    {
```

```
      "Sid": "DenySSOCreateInstance",
      "Effect": "Deny",
      "Action": ["sso:CreateInstance"],
      "Resource": "*"
    },
    {
      "Sid": "DenyRequestSpotInstances",
      "Effect": "Deny",
      "Action": ["ec2:RequestSpotInstances"],
      "Resource": "*"
    },
    {
      "Sid": "DenySpotMarketType",
      "Effect": "Deny",
      "Action": ["ec2:RunInstances"],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "ec2:InstanceMarketType": "spot"
        }
      }
    },
    {
      "Sid": "DenyReservedInstanceAndCapacityPurchases",
      "Effect": "Deny",
      "Action": [
        "athena:CreateCapacityReservation",
        "cloudfront:CreateSavingsPlan",
        "dynamodb:PurchaseReservedCapacityOfferings",
        "ec2:AllocateHosts",
        "ec2:CreateCapacityReservation",
        "ec2:CreateCapacityReservationBySplitting",
        "ec2:CreateCapacityReservationCancellationQuote",
        "ec2:CreateCapacityReservationFleet",
        "ec2:PurchaseHostReservation",
        "ec2:PurchaseReservedInstancesOffering",
        "ec2:PurchaseScheduledInstances",
        "elasticache:PurchaseReservedCacheNodesOffering",
        "es:PurchaseReservedElasticsearchInstanceOffering",
        "es:PurchaseReservedInstanceOffering",
        "glacier:PurchaseProvisionedCapacity",
        "memorydb:PurchaseReservedNodesOffering",
        "rds:PurchaseReservedDBInstancesOffering",
        "redshift:AcceptReservedNodeExchange",
```

```
        "redshift:PurchaseReservedNodeOffering",
        "sagemaker:CreateReservedCapacity",
        "savingsplans:CreateSavingsPlan"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyExportableCertificateRequests",
      "Effect": "Deny",
      "Action": ["acm:RequestCertificate"],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "acm:AllowExport": "true"
        }
      }
    }
  ]
}
```

Service control policies for the Paid Plan for projects

The following service control policy controls access to the AWS services available in the Paid Plan of our new AWS experience. This policy cannot be modified. To access additional services, activate advanced features. For more information, see Activate advanced AWS features.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AllowPaidTierServices",
      "Effect": "Deny",
      "NotAction": [
        "a2c:*",
        "access-analyzer:CheckAccessNotGranted",
        "access-analyzer:CheckNoNewAccess",
        "access-analyzer:CheckNoPublicAccess",
        "access-analyzer:ValidatePolicy",
        "account:*",
        "acm:*",
        "aco-automation:*",
        "action-recommendations:*",
        "activate:*",
```

```
        "aidevops:*",
        "aiops:*",
        "airflow-serverless:*",
        "amplify:*",
        "amplifybackend:*",
        "amplifyuibuilder:*",
        "aoss:*",
        "apigateway:*",
        "app-integrations:*",
        "appconfig:*",
        "appfabric:*",
        "appflow:*",
        "application-autoscaling:*",
        "application-signals-mcp:*",
        "application-signals:*",
        "application-transformation:*",
        "applicationinsights:*",
        "appmesh:*",
        "appsync:*",
        "aps:*",
        "arc-zonal-shift:*",
        "artifact:*",
        "athena:*",
        "autoscaling-plans:*",
        "autoscaling:*",
        "aws-external-anthropic:*",
        "aws-marketplace:*",
        "b2bi:*",
        "backup-gateway:*",
        "backup-search:*",
        "backup-storage:*",
        "backup:*",
        "batch:*",
        "bcm-pricing-calculator:*",
        "bedrock-agentcore:*",
        "bedrock-mantle:*",
        "bedrock-websearch:*",
        "bedrock:*",
        "budgets:*",
        "ce:*",
        "chatbot:*",
        "cloud9:*",
        "cloudformation:*",
        "cloudfront-keyvaluestore:*",
```

```
        "cloudfront:*",
        "cloudshell:*",
        "cloudtrail-data:*",
        "cloudtrail:*",
        "cloudwatch:*",
        "codeartifact:*",
        "codebuild:*",
        "codecommit:*",
        "codeconnections:*",
        "codedeploy-commands-secure:*",
        "codedeploy:*",
        "codeguru-profiler:*",
        "codeguru-reviewer:*",
        "codeguru-security:*",
        "codepipeline:*",
        "codestar-connections:*",
        "codestar-notifications:*",
        "cognito-identity:*",
        "cognito-idp:*",
        "cognito-sync:*",
        "compute-optimizer:*",
        "config:*",
        "consoleapp:*",
        "consolidatedbilling:*",
        "cost-optimization-hub:*",
        "customer-verification:*",
        "databrew:*",
        "dataexchange:*",
        "datapipeline:*",
        "datasync:*",
        "datazone:*",
        "dbqms:*",
        "devops-guru:*",
        "dlm:*",
        "docdb-elastic:*",
        "ds-data:*",
        "ds:*",
        "dsql:*",
        "dynamodb:*",
        "ebs:*",
        "ec2-instance-connect:*",
        "ec2:*",
        "ec2messages:*",
        "ecr-public:*",
```

```
        "ecr:*",
        "ecs-mcp:*",
        "ecs:*",
        "eks-auth:*",
        "eks-mcp:*",
        "eks:*",
        "elasticache:*",
        "elasticbeanstalk:*",
        "elasticfilesystem:*",
        "elasticloadbalancing:*",
        "elasticmapreduce:*",
        "emr-containers:*",
        "emr-serverless:*",
        "entityresolution:*",
        "es:*",
        "events:*",
        "evidently:*",
        "execute-api:*",
        "finops-agent:*",
        "firehose:*",
        "fis:*",
        "fms:*",
        "forecast:*",
        "freertos:*",
        "freetier:*",
        "geo-maps:*",
        "geo-places:*",
        "geo-routes:*",
        "geo:*",
        "glacier:*",
        "glue:*",
        "groundtruthlabeling:*",
        "health:*",
        "iam:*",
        "identitystore-auth:*",
        "identitystore:*",
        "imagebuilder:*",
        "internetmonitor:*",
        "invoicing:*",
        "iot-device-tester:*",
        "iot:*",
        "iotdeviceadvisor:*",
        "iotfleethub:*",
        "iotjobsdata:*",
```

```
        "iotmanagedintegrations:*",
        "iotsitewise:*",
        "iottwinmaker:*",
        "iotwireless:*",
        "kafka-cluster:*",
        "kafka:*",
        "kafkaconnect:*",
        "kinesis:*",
        "kinesisanalytics:*",
        "kinesisvideo:*",
        "kms:*",
        "lakeformation:*",
        "lambda:*",
        "launchwizard:*",
        "lex:*",
        "license-manager-linux-subscriptions:*",
        "license-manager-user-subscriptions:*",
        "license-manager:*",
        "logs:*",
        "managedblockchain-query:*",
        "managedblockchain:*",
        "mapcredits:*",
        "memorydb:*",
        "mgn:*",
        "migrationhub-orchestrator:*",
        "migrationhub-strategy:*",
        "mpa:*",
        "mq:*",
        "neptune-db:*",
        "neptune-graph:*",
        "network-firewall:*",
        "networkflowmonitor:*",
        "networkmanager-chat:*",
        "networkmanager:*",
        "networkmonitor:*",
        "notifications-contacts:*",
        "notifications:*",
        "nova-act:*",
        "oam:*",
        "observabilityadmin:*",
        "opensearch:*",
        "organizations:*",
        "osis:*",
        "payment-cryptography:*",
```

```
        "personalize:*",
        "pi:*",
        "pipes:*",
        "polly:*",
        "pricing:*",
        "q:DeleteConversation",
        "q:GenerateCodeFromCommands",
        "q:GenerateCodeRecommendations",
        "q:GetConversation",
        "q:GetIdentityMetadata",
        "q:GetTroubleshootingResults",
        "q:ListConversations",
        "q:PassRequest",
        "q:SendEvent",
        "q:SendMessage",
        "q:StartConversation",
        "q:StartTroubleshootingAnalysis",
        "q:StartTroubleshootingResolutionExplanation",
        "q:UpdateConversation",
        "q:UpdateTroubleshootingCommandResult",
        "q:UsePlugin",
        "ram:*",
        "rbin:*",
        "rds-data:*",
        "rds-db:*",
        "rds:*",
        "redshift-data:*",
        "redshift-serverless:*",
        "redshift:*",
        "rekognition:*",
        "resource-explorer-2:*",
        "resource-explorer:*",
        "resource-groups:*",
        "rhelkb:*",
        "route53-recovery-cluster:*",
        "route53-recovery-control-config:*",
        "route53-recovery-readiness:*",
        "route53:*",
        "route53domains:*",
        "route53globalresolver:*",
        "route53profiles:*",
        "route53resolver:*",
        "rum:*",
        "s3-object-lambda:*",
```

```
        "s3:*",
        "s3express:*",
        "s3files:*",
        "s3tables:*",
        "s3vectors:*",
        "sagemaker-data-science-assistant:*",
        "sagemaker-geospatial:*",
        "sagemaker-mlflow:*",
        "sagemaker-unified-studio-mcp:*",
        "sagemaker:*",
        "scheduler:*",
        "schemas:*",
        "secretsmanager:*",
        "securityagent:*",
        "serverlessrepo:*",
        "servicecatalog:Describe*",
        "servicecatalog:List*",
        "servicecatalog:ProvisionProduct",
        "servicecatalog:ScanProvisionedProducts",
        "servicecatalog:SearchProducts",
        "servicecatalog:SearchProvisionedProducts",
        "servicecatalog:TerminateProvisionedProduct",
        "servicecatalog:UpdateProvisionedProduct",
        "servicediscovery:*",
        "serviceextract:*",
        "servicequotas:*",
        "ses:*",
        "signer:*",
        "signin:*",
        "sms-voice:*",
        "sns:*",
        "social-messaging:*",
        "sqlworkbench:*",
        "sqs:*",
        "ssm-contacts:*",
        "ssm-guiconnect:*",
        "ssm-incidents:*",
        "ssm-quicksetup:*",
        "ssm-sap:*",
        "ssm:*",
        "ssmmessages:*",
        "sso-directory:*",
        "sso-oauth:*",
        "sso:*",
```

```
        "states:*",
        "storagegateway:*",
        "sts:*",
        "support:*",
        "swf:*",
        "synthetics:*",
        "tag:*",
        "tax:*",
        "textract:*",
        "timestream-influxdb:*",
        "timestream:*",
        "transcribe:*",
        "translate:*",
        "ts:*",
        "uxc:*",
        "verifiedpermissions:*",
        "vpce:*",
        "waf-regional:*",
        "waf:*",
        "wafv2:*",
        "wellarchitected:*",
        "xray:*"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyEC2CapacityBlock",
      "Effect": "Deny",
      "Action": ["ec2:PurchaseCapacityBlock", "ec2:PurchaseCapacityBlockExtension"],
      "Resource": "*"
    },
    {
      "Sid": "DenyRestrictedAgreements",
      "Effect": "Deny",
      "Action": ["artifact:AcceptAgreement", "artifact:AcceptNdaForAgreement"],
      "Resource": [
        "arn:aws:artifact:::agreement/agreement-9c1kBcYznTkcpRIm",
        "arn:aws:artifact:::agreement/agreement-y03aUwMAEorHtqjv",
        "arn:aws:artifact:::agreement/agreement-bexgr7sjvXAW4Gxu",
        "arn:aws:artifact:::agreement/agreement-HZTdNwJuqOKLReXC"
      ]
    },
    {
      "Sid": "DenyCloudWAN",
```

```
      "Effect": "Deny",
      "Action": [
        "networkmanager:CreateCoreNetwork",
        "networkmanager:CreateConnectAttachment",
        "networkmanager:CreateConnectPeer",
        "networkmanager:CreateVpcAttachment",
        "networkmanager:CreateSiteToSiteVpnAttachment",
        "networkmanager:CreateTransitGatewayPeering",
        "networkmanager:CreateDirectConnectGatewayAttachment",
        "networkmanager:PutCoreNetworkPolicy",
        "networkmanager:ExecuteCoreNetworkChangeSet"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyTransitGatewayAndVPN",
      "Effect": "Deny",
      "Action": [
        "ec2:CreateTransitGateway",
        "ec2:CreateTransitGatewayVpcAttachment",
        "ec2:CreateTransitGatewayConnect",
        "ec2:CreateTransitGatewayConnectPeer",
        "ec2:CreateTransitGatewayMulticastDomain",
        "ec2:CreateTransitGatewayPeeringAttachment",
        "ec2:CreateTransitGatewayPolicyTable",
        "ec2:CreateTransitGatewayPrefixListReference",
        "ec2:CreateTransitGatewayRoute",
        "ec2:CreateTransitGatewayRouteTable",
        "ec2:CreateTransitGatewayRouteTableAnnouncement",
        "ec2:CreateClientVpnEndpoint",
        "ec2:CreateClientVpnRoute",
        "ec2:AuthorizeClientVpnIngress",
        "ec2:AssociateClientVpnTargetNetwork",
        "ec2:CreateVpnConnection",
        "ec2:CreateVpnConnectionRoute",
        "ec2:CreateVpnGateway",
        "ec2:CreateCustomerGateway"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenySageMakerGroundTruth",
      "Effect": "Deny",
      "Action": [
```

```
        "sagemaker:CreateLabelingJob",
        "sagemaker:CreateWorkteam",
        "sagemaker:CreateWorkforce",
        "sagemaker:CreateHumanTaskUi"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyEC2Fleet",
      "Effect": "Deny",
      "Action": ["ec2:CreateFleet"],
      "Resource": "*"
    },
    {
      "Sid": "DenyEC2BYOLAndLicenseConfigurations",
      "Effect": "Deny",
      "Action": ["ec2:RunInstances", "ec2:StartInstances"],
      "Resource": "arn:aws:ec2:*:*:instance/*",
      "Condition": {
        "StringEquals": {
          "ec2:Tenancy": "host"
        }
      }
    },
    {
      "Sid": "DenyEC2InstanceStoreRootDevice",
      "Effect": "Deny",
      "Action": ["ec2:RunInstances"],
      "Resource": "arn:aws:ec2:*:*:instance/*",
      "Condition": {
        "StringEquals": {
          "ec2:RootDeviceType": "instance-store"
        }
      }
    },
    {
      "Sid": "DenyLicenseManagerAssociation",
      "Effect": "Deny",
      "Action": ["license-manager:CreateLicenseConfiguration"],
      "Resource": "*"
    },
    {
      "Sid": "DenyIAMRestrictedActions",
      "Effect": "Deny",
```

```
      "Action": [
        "iam:*Alias*",
        "iam:*Organizations*",
        "iam:*Provider*",
        "iam:SetSecurityTokenServicePreferences",
        "iam:*LoginProfile*",
        "iam:CreateGroup",
        "iam:AttachGroupPolicy",
        "iam:AddUserToGroup",
        "iam:PutGroupPolicy",
        "iam:UpdateGroup",
        "iam:UpdateServerCertificate",
        "iam:UpdateSigningCertificate",
        "iam:UploadServerCertificate",
        "iam:UploadSigningCertificate",
        "iam:TagServerCertificate",
        "iam:UntagServerCertificate"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenySSOCreateInstance",
      "Effect": "Deny",
      "Action": ["sso:CreateInstance"],
      "Resource": "*"
    },
    {
      "Sid": "DenyRequestSpotInstances",
      "Effect": "Deny",
      "Action": ["ec2:RequestSpotInstances"],
      "Resource": "*"
    },
    {
      "Sid": "DenySpotMarketType",
      "Effect": "Deny",
      "Action": ["ec2:RunInstances"],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "ec2:InstanceMarketType": "spot"
        }
      }
    },
    {
```

```
      "Sid": "DenyReservedInstanceAndCapacityPurchases",
      "Effect": "Deny",
      "Action": [
        "athena:CreateCapacityReservation",
        "cloudfront:CreateSavingsPlan",
        "dynamodb:PurchaseReservedCapacityOfferings",
        "ec2:AllocateHosts",
        "ec2:CreateCapacityReservation",
        "ec2:CreateCapacityReservationBySplitting",
        "ec2:CreateCapacityReservationCancellationQuote",
        "ec2:CreateCapacityReservationFleet",
        "ec2:PurchaseHostReservation",
        "ec2:PurchaseReservedInstancesOffering",
        "ec2:PurchaseScheduledInstances",
        "elasticache:PurchaseReservedCacheNodesOffering",
        "es:PurchaseReservedElasticsearchInstanceOffering",
        "es:PurchaseReservedInstanceOffering",
        "glacier:PurchaseProvisionedCapacity",
        "memorydb:PurchaseReservedNodesOffering",
        "rds:PurchaseReservedDBInstancesOffering",
        "redshift:AcceptReservedNodeExchange",
        "redshift:PurchaseReservedNodeOffering",
        "sagemaker:CreateReservedCapacity",
        "savingsplans:CreateSavingsPlan"
      ],
      "Resource": "*"
    },
    {
      "Sid": "DenyExportableCertificateRequests",
      "Effect": "Deny",
      "Action": ["acm:RequestCertificate"],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "acm:AllowExport": "true"
        }
      }
    }
  ]
}
```

Resource control policies for projects

The following is the resource control policy for the resources in the project. This policy cannot be modified:

```
{
  "Version" : "2012-10-17",
  "Statement" : [
    {
      "Sid" : "DenyAnyoneOutsideMyOrgAndAWS",
      "Effect" : "Deny",
      "Principal" : "*",
      "Action" : [
        "aoss:*",
        "appconfig:*",
        "appstream:*",
        "autoscaling:*",
        "codebuild:*",
        "codecommit:*",
        "cognito-identity:*",
        "cognito-idp:*",
        "comprehend:*",
        "comprehendmedical:*",
        "dax:*",
        "dynamodb:*",
        "ecr:*",
        "health:*",
        "kinesisvideo:*",
        "kms:*",
        "logs:*",
        "s3:*",
        "secretsmanager:*",
        "sqs:*",
        "sts:*",
        "support:*",
        "textract:*",
        "transcribe:*",
        "translate:*"
      ],
      "Resource" : "*",
      "Condition" : {
        "BoolIfExists" : {
          "aws:PrincipalIsAWSService" : "false"
```

```
        },
        "Null" : {
          "aws:PrincipalARN" : "false"
        },
        "StringNotEqualsIfExists" : {
          "aws:PrincipalOrgID" : "${aws:ResourceOrgID}"
        }
      }
    },
    {
      "Sid" : "DenyAWSWhenItsNotMyOrgsSourceAccount",
      "Effect" : "Deny",
      "Principal" : "*",
      "Action" : [
        "aoss:*",
        "appconfig:*",
        "appstream:*",
        "autoscaling:*",
        "codebuild:*",
        "codecommit:*",
        "cognito-identity:*",
        "cognito-idp:*",
        "comprehend:*",
        "comprehendmedical:*",
        "dax:*",
        "dynamodb:*",
        "ecr:*",
        "health:*",
        "kinesisvideo:*",
        "kms:*",
        "logs:*",
        "s3:*",
        "secretsmanager:*",
        "sqs:*",
        "sts:*",
        "support:*",
        "textract:*",
        "transcribe:*",
        "translate:*"
      ],
      "Resource" : "*",
      "Condition" : {
        "Null" : {
          "aws:SourceAccount" : "false"
```

```
        },
        "StringNotEquals" : {
          "aws:SourceOrgID" : "${aws:ResourceOrgID}"
        }
      }
    }
  ]
}
```

Switch between projects

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can only manage AWS resources for one project at a time. You can always tell which project you are in based on the name in the upper right of the AWS console.

Switch between projects for active sessions from the AWS Management Console

To switch between projects for active sessions from the AWS Management Console

1. Open the AWS Management Console.
2. Choose your project name on the upper right of the AWS Management Console.
3. Choose **Switch project**.
4. Under **Active sessions**, choose an active session. An active session is a session that's been signed in for up to 12 hours. The active session time can't be changed.
5. You'll be redirected to the AWS console home page for your new project.

Switch between projects in AWS Settings

To switch between projects in AWS Settings

1. Open AWS Settings at https://settings.aws.com.

2. In the main navigation pane, choose the project you want to access.
3. In the navigation bar on the upper right, choose **Create and manage cloud infrastructure**.

   You'll be redirected back to the AWS Management Console for your new project.

Switch between projects from the AWS Management Console

To switch between projects from the AWS Management Console

1. Open the AWS Management Console.
2. Choose **Manage projects**.
3. In AWS Settings, choose the project you want to access.
4. In the navigation bar on the upper right, choose **Create and manage cloud infrastructure**.

   You'll be redirected back to the AWS Management Console for your new project.

Invite team members to collaborate in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can give team members access to your projects. When you share a project with a team member, you cannot control which AWS services they have access to. Both you and the team member have admin access to the project. This means, the team member has permissions to access all AWS services available for your project. This includes creating access controls for services and applications. However, they don't have permissions to perform administrative or financial actions, such as deleting your project or modifying a spend limit. To view which team member has modified an AWS service, see Monitor your project.

In AWS Settings, you and your team members have different levels of access. To view the AWS managed policy that is applied to a team member, see AWSManagedSettingsReadOnlyAccess. To view the AWS managed policy that is applied to a project owner, see AWSManagedSettingsAdminAccess.

Consider the following when adding a team member to your project.

Considerations for team members

A team member can do the following actions:

- View, create, edit, delete, and modify AWS resources in a project.
- Access AWS Settings.
- View spend limits.
- Request a quota increase for an AWS service.

A team member can't do the following actions:

- Leave a project.
- Take any root actions. For more information, see AWS actions for project owners.
- Invite team members to projects.
- Manage any team member's permissions.
- Remove any team member from a project.
- Change the name of a project.
- Set up or update payment methods.
- View or update billing details, including address or phone number.
- View or update tax information.
- Upgrade from Free Tier to Paid Plan.
- Set or modify spend limits per projects.
- View cost trends for each project.
- View total AWS spending across all projects.
- View and redeem non-Free Tier promotional credits.
- View remaining credits and expiration dates for all credit types.
- View full credit history including earned, applied, and expired credits.
- View how credits apply across projects.
- View outstanding balance.

- Execute payments to AWS.
- Download invoices or view past transactions.
- Validate that AWS charges are correct, including credits applied.
- Receive management account-level billing communications.
- Get help with billing issues at the organization level.
- Request a quota increase for account and billing quotas.

How team members access your AWS resources

Our new AWS experience creates a service-linked role for the organization that contains all your projects that you cannot modify. This role periodically verifies that the internal records of your team members stay in sync with your actual team members.

The **AWSServiceRoleForAccountAccessManager** service-linked role trusts the following services to assume the role:

- `account-access.amazonaws.com`

The role permissions policy named `AccountAccessManagerServiceRolePolicy` allows account access manager to complete the following actions on the specified resources:

- Action: `organizations:ListAccounts`, `organizations:ListAWSServiceAccessForOrganization`, `organizations:DescribeAccount` on `"arn:aws:organizations::*:account/o-*/*"`

For more information, see AWS Account Access Manager.

Invite team members to collaborate in AWS Settings

To invite team members, you send them an invitation from the Team page in the AWS Settings. Team members have 14 days to accept their invitation. Team members must create a AWS Builder ID to access your project. They can do this using Sign up for AWS (new).

Invite team members to collaborate in AWS Settings

1. Open AWS Settings at https://settings.aws.com.

2. In the main navigation pane, choose **Team**.
3. Choose **Invite new team member**.
4. For **Email**, enter an email address or a list of email addresses. Separate the email addresses with commas (,) or semicolons (;).
5. Choose one or more projects to share with your new team members.
6. Choose **Send invitation**.

If your team members don't get the invitation, ask them to check their spam and trash folders.

Invite a team member to a project in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can invite a team member to access a project. For information on what team members can do in a project, see Invite team members to collaborate in AWS Settings.

To invite a team member to a project

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose the project you want to access.
3. For **Members**, choose **Invite new team member**.
4. For **Email**, enter an email address or a list of email addresses. Separate the email addresses with commas (,) or semicolons (;).
5. Choose a project to share with your new team members.
6. Choose **Send invitation**.

If your team members don't get the invitation, ask them to check their spam and trash folders.

Remove a team member from a project in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can remove a team member from a project at any time. Your team members can't do this on their own. When you remove a team member, they will no longer have access to that project. Any resources that your team member created remain in your project and you must manage them.

To remove a team member from a project

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose the project you want to access.
3. In the **Members** tab, select the member you want to remove from the project with, and then choose **Remove**.
4. Confirm your choice and choose **Remove**.

The team member is still part of your team and you can add them to other projects. To completely remove a team member from AWS Settings, see Remove a team member in AWS Settings.

Remove a team member in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can use the Team page to remove a team member from all projects. They will automatically lose access to any projects. Any resources the team member created will not be deleted.

To remove a team member

1. Open AWS Settings at https://settings.aws.com.

2. In the main navigation pane, choose **Team**.
3. Select the checkbox for the team member you want to remove.
4. Choose **Remove**.
5. Confirm your choice and choose **Remove**.

Revoke an invitation to a team member in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can revoke a pending invitation. The invitation will remain in AWS Settings for 14 days. When you revoke a pending invitation, the access link in the invitation email will be deactivated. You can't revoke an accepted invitation, instead remove the team member. For more information, see Remove a team member in AWS Settings.

To revoke an invitation

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Team**.
3. Select the checkbox for the team member that has a pending invitation.
4. Choose **Revoke**.
5. Confirm your choice and choose **Revoke invitation**.

Resend an invitation to a team member in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can resend a team member an invite for a revoked, expired, or declined invitation. When you resend an invitation, your team member receives a new email with an expiration date 14 days from when you resend the invitation.

To resend an invite

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Team**.
3. Select the checkbox for the team member that has a revoked, expired, or declined invitation.
4. Choose **Resend**.
5. Confirm your choice and choose **Resend invitation**.

Upgrade your account in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you sign up with Sign up for AWS (new), your accounts are in the Free Tier. In the Free Tier, you receive USD $100 sign up credit and earn up to $100 in additional credits. When you either run out of credits or have an account that is over 6 months, you must upgrade to a Paid Plan. You can also upgrade to a Paid Plan at any time. When you upgrade to a Paid Plan, you have access to the services listed in AWS services included in the Paid Plan of our new AWS experience.

Considerations for upgrading your account

Consider the following when you upgrade to a Paid Plan:

- You cannot upgrade individual projects. All your projects are upgraded when you upgrade your account.
- When you upgrade, you will have access to Always Free services and short-term trial offers.
- You are eligible for other promotional credits and discounts.
- You must pay for charges that exceed the credit balance.

- You can create a spend limit to set the most you'll pay per month for a project. For more information, see Create a spend limit in AWS Settings.
- You cannot designate any projects as HIPAA or SEC compliant.
- Your account doesn't close when credits are depleted.
- This does not upgrade you to Sign up for AWS (advanced). None of the features that are only supported for accounts created using Sign up for AWS (advanced) will become available when you upgrade your account. However, you can Activate advanced AWS features for your account.
- You cannot reverse the upgrade process.

Upgrade your account in AWS Settings

To upgrade your account

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. Choose **Upgrade**.
4. Choose **Get started**.
5. Add your payment method.

   Based on your location, you might need to provide some tax information.
6. Choose **Upgrade account**.
7. Choose **Continue** to return to building or choose **Set spend limits** to set a spend limit. For more information on spend limits, see Create a spend limit in AWS Settings.
8. Choose **Go to console**.

Your upgrade is complete.

Create a spend limit in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

What is a spend limit?

Spend limits enable developers to use AWS with confidence that they can operate within a predictable and controlled budget. Use a spend limit to set the most you'll pay per month for a project. Spend limits are at the project level. You can have spend limits on some of your projects and have other projects without spend limits.

If your usage in a project reaches its limit, AWS pauses that project which stops its resources so your costs stay within it. Spend limits are designed for experimentation, learning, and sandbox workloads. They can be used for production environments when it is acceptable to have a brief pause of your resources in cases when your applications incur unexpected costs.

Spend limits are visible to anyone with project access, but can only be managed by project owners. You must be on a Paid Plan to create a spend limit. For more information, see Upgrade your account in AWS Settings.

Spend limits are a type of AWS Budgets.

Set your spend limit

Your spend limit is the ceiling on your project's pre-tax costs, not a fee. When you set a spend limit, the minimum allowed value is the greater of $20 or a conservative estimate of your likely spend.

Why a minimum exists: Hitting a spend limit is a disruptive experience. The minimum exists to make it unlikely that normal variations in your usage will lead to hitting the spend limit.

How the conservative estimate is calculated: The estimate is based on your activity month to date, but it's not a strict linear extrapolation. AWS looks at your spend to date and the resources you currently have running. AWS also uses your activity from the prior month — if that was high, it becomes the minimum.

If you have many resources running: The conservative estimate of your likely spend will be high. If you want to set a lower spend limit, you need to stop those resources first.

If you are reactivating a paused project: The conservative estimate of your likely spend is your spend to date during the month plus any usage that was not charged.

> **Note**
>
> Spend limits exclude credits and apply to pre-tax charges.

Notifications

You'll receive notifications when actual costs reach 50%, 75%, and 90% of your limit, or when you are on track to reach or exceed your spend limit in the next 10 days.

Customize your spend limit settings with optional early cost controls

Your spend limit keeps your bill from growing past the amount you set. If your usage reaches it, AWS pauses your resources to protect you. These optional controls act earlier and more gently, so you stay well below the limit.

Stop new resource creation

About 7 days before you would reach your limit, new resources stop launching. This is achieved through application of a service control policy that is managed by AWS. The service control policy depends on the service. To view all service control policies that might be applied to your project, see Service control policies for spend limits.

All your resources that are already running stay running. In most cases, nothing you have built is affected. However, in some cases this could lead to a disruptive experience. For example, if you have enabled auto-scaling, and if conditions are met to execute this action, new instances won't launch.

Pause idle resources

About 5 days before you would reach your limit, idle resources are paused. Your active apps keep running. Idle resources are paused based on recommendations by AWS Compute Optimizer. For more information, see View idle recommendations.

You can also view your idle resources recommendations by logging into your project and viewing savings opportunities. For more information, see Cost Optimization Hub. At this time, as part of your spend limits experience we will automatically pause EC2, RDS and SageMaker endpoints.

Why we pause an EC2 instance: We pause EC2 instances if the peak CPU utilization is below 5% and your network I/O is less than 5 MB per day over the last 14 days.

If you have a G or P instance type, there's a different idle criterion. G or P instance types will be paused if the following is true over the last 14 days:

- GPU isn't actively working for more than 99% of the lookback period

- GPU encoder isn't used for 99% or more of the instance's runtime
- GPU memory usage at instance level is less than 5%
- CPU maximum utilization is less than 5%
- Network utilization is less than 5 MB/day

Why we pause an RDS database: We pause RDS for MySQL and RDS for PostgreSQL if the database instance is not a read replica and has the following over the past 14 days:

- No database connections
- Low CPU usage
- Low read/write activity

Why we pause an idle SageMaker endpoint: We pause SageMaker endpoints if the endpoint had zero invocations in the past 14 days.

Pause top cost drivers

About 4 days before you would reach your limit, your highest cost incurring active resources are paused. This can be a disruptive experience. At this time, we select the resources from these five services: EC2, RDS, Lambda, Bedrock, and SageMaker. This opt-in control might prevent you from incurring costs if you have a runaway Lambda or unexpected Bedrock spike.

How we pause a top cost driver EC2 instance: When an EC2 instance is identified as a top cost driver, AWS terminates the instance to eliminate compute costs. Before termination, AWS creates a snapshot of each attached EBS volume to preserve your data. Any associated Elastic IP addresses are also released.

After the instance is terminated:

- All compute and EBS volume costs are eliminated
- Your data is preserved in snapshots, which incur a small storage cost
- Elastic IP charges stop

To restore your workload, you can launch a new instance from the saved snapshots. If you no longer need the data, you can delete the snapshots to stop snapshot storage charges.

How we pause a top cost driver RDS database: When an RDS database instance is identified as a top cost driver, AWS stops the instance. Stopping the instance eliminates compute charges while preserving your data.

While the instance is stopped:

- No compute charges are incurred
- Storage and provisioned IOPS charges continue
- Automated backups continue
- RDS automatically restarts the instance after 7 days — AWS will stop it again if the spend limit is still at risk

To restore your database, start the instance from AWS Settings or the RDS console. Your data and configuration remain intact.

How we pause a top cost driver Lambda function: When a Lambda function is identified as a top cost driver, AWS disables its event source mappings and triggers to prevent further invocations. If the function has provisioned concurrency configured, AWS also removes it to eliminate those charges.

After the function is paused:

- No invocation charges are incurred
- The function code, configuration, and all associated resources remain intact
- No data is lost

To restore your function, re-enable the event source mappings and triggers. If you had provisioned concurrency, you'll need to reconfigure it.

How we pause a top cost driver Bedrock model: When an Amazon Bedrock provisioned model is identified as a top cost driver, AWS deletes the provisioned throughput. There is no pause option for provisioned throughput — deletion is the only way to stop charges.

After the provisioned throughput is deleted:

- Provisioned throughput charges stop immediately
- If you have a custom fine-tuned model, the model weights are deleted

- Your original training data in Amazon S3 is not affected

To restore a custom model, you'll need to retrain it from your S3 training data. For provisioned throughput on foundation models, you can create a new provisioned throughput allocation.

What happens if you reach your limit

AWS pauses your project and stops all resources. Your data is preserved.

To reactivate your project, increase your spend limit from AWS Settings. After reactivation, some resources may need to be manually restarted.

> **Important**
>
> If you take no action within 90 days of your project being paused, AWS permanently deletes your project data.

Other important things to note

- If you use Amazon Route 53, you can only purchase one domain every 24 hours. You must ensure that domain costs are within spend limits, otherwise, your domain purchase won't go through. In such cases, you must increase spend limits before making a purchase.
- You can apply spend limits to up to 10 projects.

Create a spend limit in AWS Settings

To create a spend limit in AWS Settings

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. In **Cost by project**, for **Spend limit**, choose **Set limit** for a project.
4. You can either choose the recommended spend limit or a custom limit.
5. If you choose a custom limit, enter the custom amount.
6. Choose **Set spend limit**.

7. To customize your spend limit with early cost controls, you can select **Stop new resource launches**, **Pause idle resources**, or **Pause top cost drivers**.
8. Choose **Save early cost controls**.

The spend limit will take effect immediately.

Update a spend limit in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

The project owner can update a spend limit in AWS Settings at any time. You can increase or decrease your spend limit, or change the optional early cost controls. If AWS has blocked new resource launches or paused resources because you are approaching your limit, you can increase your spend limit. If your project has been paused because you reached your limit, increase your spend limit to reactivate your project. For more information about how spend limits work, see Create a spend limit in AWS Settings.

Update a spend limit in AWS Settings

To update a spend limit

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. In **Cost by project**, for **Spend limit**, choose the pencil icon.
4. For **Monthly spend limit (USD)**, enter a new spend limit.

   We provide a recommendation based on your current spend.
5. To modify which protections you want on your project, select **Stop new resource launches**, **Pause idle resources**, or **Pause top cost drivers**.
6. Choose **Save settings**.

The spend limit changes will take effect immediately.

Remove a spend limit in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

The project owner can remove a spend limit in AWS Settings at any time. When you remove a spend limit, AWS will no longer monitor your costs or take any early cost controls on your project. For more information about how spend limits work, see Create a spend limit in AWS Settings.

Remove a spend limit in AWS Settings

To remove a spend limit

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. In **Cost by project**, for **Spend limit**, choose the pencil icon.
4. Choose **Remove limit**.
5. Confirm your choice and choose **Remove**.

Your spend limit will be removed immediately.

Free Tier in our new AWS experience

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can use AWS Free Tier to explore AWS services without cost commitments. If you are new to AWS, you receive USD $100 in credits after you create an account, regardless of your account plan. You can also earn up to an additional USD $100 across all your projects in credits by completing activities. For more information about earning additional credits, see Earn additional credits. In addition, you have access to over 30 always free AWS services that offer monthly free usage limits.

The Free account plan is ideal for customers experimenting with AWS services and building proof of concepts at no cost for up to six months. You will not incur any charges during this period until you upgrade to a paid account plan. Your free account plan ends after six months or when your credits are fully used - whichever occurs first. Additionally, free account plans don't have access to certain AWS services that would rapidly consume the entire AWS Free Tier credit amount, or hardware purchases. For a list of eligible services on the free account plan, see Supported AWS services for Sign up for AWS (new).

The Paid account plan is ideal for building production applications that scale beyond the initial credit amount. When your usage exceeds your credit balance or when you use a service where credits don't apply, you pay standard pay-as-you-go pricing.

All projects that you own are under the same plan type. If you upgrade to the Paid account plan, all of your projects have access to additional services, and you pay standard pay-as-you-go pricing for all projects. For more information, see Upgrade your account in AWS Settings.

You can use the Cost Management Console to track your usage across all AWS services offering Free Tier for your project.

Track your Free Tier usage

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

AWS Free Tier usage alerts automatically notifies you over email when you exceed 85 percent of your Free Tier limit for each service. For additional tracking, you can configure to track your usage to 100% of the Free Tier limit by setting a zero spend budget. You can also filter your budget to track individual services.

You can track your AWS Free Tier usage in the following ways.

Using AWS Settings

You can monitor your free account plan expiration date, credit balance, and days remaining in AWS Settings. For more information, see View and redeem credits in AWS Settings. You also receive periodic email alerts regarding your credit balance and when you are approaching the end of your free account plan period.

If you have a paid plan, you can monitor your credit balance and expiration date on the credits page in AWS Settings. For more information about upgrading to a paid plan, see Upgrade your account in AWS Settings.

Using the Billing and Cost Management Console

You can use the Billing and Cost Management Console to track your free tier usage across each project. The Billing and Cost Management Console shows a summary of your month-to-date usage and any service offerings at or above usage limits. For each free tier offer in use, you can view the following:

- The service
- The AWS Free Tier usage limit
- Your current usage
- Your forecasted usage
- Your month-to-date (MTD) actual usage
- Your forecasted month-to-date (MTD) usage

The following procedure shows you how to use the Billing and Cost Management Console to sort your Free Tier usage by your month-to-date actual usage.

To sort your Free Tier usage

1. Open the Billing and Cost Management console at https://console.aws.amazon.com/cost-management/.
2. Choose **Free Tier**.
3. To sort by month-to-date actual usage, choose **MTD actual usage %**.

Earn additional credits

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

If you are a new AWS customer, you can earn an additional USD $100 in credits regardless of your account plan by completing activities. These activities assist your learning of AWS services. You can find the activities in the Explore AWS widget on your AWS Console Home dashboard. All of these activities will incur AWS service charges that are deducted from your AWS Free Tier credits, however when you complete each activity, you receive USD $20 in credits.

Activities to complete

- Launch an instance using Amazon EC2
- Use a foundational model in the Amazon Bedrock playground
- Create a budget
- Create a web app using Lambda
- Create an Amazon RDS database

View your billing information in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can use AWS Settings to view your billing information. If you're using the Free Tier, this page shows you how many days of the Free Tier you have and your AWS credits. You can also view the cost per project. If you're using the Paid Plan, you can view your balance, current charges, and the cost per project. If you're using a spend limit, you can view information related to your limit. You can also Manage your payment method in AWS Settings and Manage tax registration in AWS Settings.

When you use the Free Plan, all usage is covered by your AWS Free Tier credits. You need to upgrade to the Paid Plan if you either exhaust your AWS Free Tier credits or after 184 days, whichever happens first. For more information about upgrading to a Paid Plan, see Upgrade your account in AWS Settings.

To view your cost by project

1. Open AWS Settings at https://settings.aws.com.

2. In the main navigation pane, choose **Billing**.
3. Under **Cost by project** you'll see your month-to-date charges and the forecasted month-to-date charges. You'll see all charges across your projects.
4. (Optional) To view a breakdown of your charges for a project under **Inspect costs**, choose **View**. You'll be redirected to the AWS Billing and Cost Management console.

Your billing data and Cost Explorer data that you see on the AWS Billing and Cost Management Console are refreshed at least once per day. The cadence when they're refreshed might differ. If you use a spend limit, this can result in differences for your month-to-date estimated charges in AWS Settings and AWS Cost Management Console.

View your billing history in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can view your billing history in AWS Settings. Your billing history is only available if you have a payment method. If you're using the Free Tier, you won't see your billing history.

You can download the invoices for each service provider that you transacted with during the billing period. This includes information such as charge type, invoice date, and total in USD. If your invoices are issued in another currency, the total in the other currency is also shown. The invoices are not split across each project.

To view your billing history

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. Next to **Balance**, choose **View billing history**. You'll be redirected to the billing history.
4. To download an invoice, select an invoice ID.

Manage your payment method in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can manage your payment method in AWS Settings. Your payment method is used to pay for AWS resource usage in all of your projects. You can only manage your payment method if you provided information when you created your account or if you upgraded your account to the Paid Plan.

To manage your payment method

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. Under **Balance**, you'll see your payment information. Choose **Manage**.
4. For **Payment method**, choose **Update**.
5. Enter your new payment method and choose **Save**.

Make a payment in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can pay your AWS bill in AWS Settings. AWS charges your default payment method automatically at the beginning of each month. If that charge doesn't process successfully, you can use the console to update your payment method and make a payment.

Before making a payment, make sure that the payment method that you want to be automatically charged in the future is set as your default payment method. If you're using a credit card, confirm that your credit card isn't expired.

To make a payment

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. Next to **Balance**, choose **View billing history**. You'll be redirected to the billing history.
4. Under **Action**, you'll see **Pay invoice** for any outstanding invoice. If there are no invoices listed, you don't need to do anything.
5. Choose **Pay invoice**.
6. Confirm your choice and choose **Pay**.

If you have questions about issues with charging your bank account or paying an overdue balance, create a case in the Support Center.

Change your preferred currency in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can change your preferred currency in AWS Settings. Your preferred currency only applies to the projects that you own. Certain billing countries restrict payment currencies. Your card issuer might charge a fee for transitions in other currencies.

To change your preferred display currency

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. Under **Balance**, you'll see your payment information. Choose **Manage**. If you don't see a balance, you cannot change the currency.
4. Choose **Set currency**.
5. For **Default payment currency**, select the payment currency.
6. Choose **Save**.

Manage tax registration in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You manage your tax registration numbers and tax exemptions in AWS Settings. Your tax information only applies to the projects that you own.

Updating and deleting tax registration numbers

Use the following steps to update or delete one or more tax registration numbers.

> **Note**
>
> If a country isn't listed in the **Tax settings** page dropdown, AWS doesn't collect tax registration for that country at this time.

To update tax registration numbers

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. Under **Balance**, you'll see your payment information. Choose **Manage**.
4. Choose **Tax settings**.
5. Modify your tax registration information.
6. Choose **Save**.

Managing your US tax exemptions

If your state is eligible, you can manage your US tax exemptions on the **Plan & billing** page in AWS Settings. The documents you upload for the exemption are reviewed by Support within 24 hours. You can only manage your tax exemptions if your account is registered as a business.

To upload or add your US tax exemption

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. Under **Balance**, you'll see your payment information. Choose **Manage**.
4. Choose **Manage tax exemption** and select **Add tax exemption**.
5. Specify your exemption type and jurisdiction.
6. Upload certificate documents.
7. Review your information, and choose **Submit**.

Within 24 hours, Support will notify you through a support case if they need additional information, or if any of your documents weren't valid.

Once the exemption is approved, you can view it under the **Tax exemption** tab with an **Active** validity period.

You will be notified through a support case contact if your exemption was rejected.

View and redeem credits in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

You can view and redeem your available credits in AWS Settings. AWS credits are automatically applied to bills to help cover costs that are associated with eligible services. For more information about eligible services, see Redeem Your AWS Promotional Credit. Credits are applied across all projects that you own until they are exhausted or they expire. You cannot select which project uses the credits that you redeem, they are applied across all projects that you own.

To view your credits

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.

3. In the **Current charges** container, you'll see your available credits. Choose your credits.

   You'll be redirected to a view of active and expired credits, or your credits and last 6 months of inactive credits, depending on your plan type.
4. To view the AWS services that can be used with your credits, choose **See list of services**.

To redeem your AWS Promotional Credits

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Billing**.
3. In the **Current charges** container, you'll see your available credits. Choose your credits.

   You'll be redirected to a view of active and expired credits, or your credits and last 6 months of inactive credits, depending on your plan type.
4. Choose **Redeem credit** to redeem your credits.
5. For **promotional code**, enter the promotional code as it appears. When you redeem the credit, you indicate that you read and agree to the AWS Promotional Credits Terms and Conditions.
6. Choose **Redeem credit**.

Manage accounts in India in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

If you sign up for AWS and choose India for your contact and billing address, your user agreement is with Amazon Web Services India Private Limited (AWS India), a local AWS seller in India. AWS India manages your billing, and your invoice total is listed in Indian rupees (INR) instead of US dollars (USD).

If your account is with AWS India, follow the procedures in this topic to manage your account. This topic explains how to sign up for an AWS India account, edit information about your AWS India account, manage customer verification, and add or edit your Permanent Account Number (PAN).

As part of the credit card verification during sign up, AWS India charges your credit card 2 INR. AWS India refunds the 2 INR after verification is done. You might be redirected to your bank as part of the verification process.

Topics

Sign up for AWS (new) with AWS India

AWS India is a local seller of AWS in India. If your contact and billing address is in India, use the following procedure to sign up.

To use Sign up for AWS (new) with AWS India

1. Go to aws.amazon.com and choose **Create account**.
2. On the banner explaining the differences between Sign up for AWS (new) and Sign up for AWS (advanced), choose **Sign up for AWS**.
3. Choose a login such as Google, Apple, GitHub, or Amazon.
4. You'll be directed to the login for your provider. After you login, you'll be redirected to AWS sign up.
5. Enter your name and choose **Continue**.

   This name will be shared with collaborators when you give them access to your projects.
6. Verify the email associated with your login by entering a verification code. When you've verified your email, choose **Continue**.
7. Enter a strong password for your account and choose **Continue**.

   AWS requires that your password meet the following conditions:

   - It must have a minimum of 8 characters and a maximum of 128 characters.
   - It must include a minimum of three of the following mix of character types: uppercase, lowercase, numbers, and ! @ # $ % ^ management portal * () <> [] {} | _+-= symbols.
   - It must not be identical to your AWS account name or email address.
8. Enter your contact details and choose **Continue**.

If your contact or billing address is based in India, in compliance with Indian Computer Emergency Response Team (CERT-In) regulations, AWS is required to collect and validate your identity information before granting you access to AWS services.

The name that you chose between your contact or billing information must exactly match the name that appears on the document you plan to use for customer verification. For example, if you plan to verify a business account using a Certificate of Incorporation, you must provide the business name that appears on the document. For a list of accepted document types, see the section called “Accepted India documents for customer verification”.

9. Enter your payment information and choose **Continue**. AWS India charges your card 2 INR as part of the verification process. AWS India refunds the 2 INR after verification is done.
10. Choose the primary purpose of your account registration.
11. Choose the ownership type that best represents the owner of the account. If you choose a company, organization, or partnership as the ownership type, enter the name of a key managerial person. The key managerial person can be a director, head of operations, or a person in charge of operations in your business.
12. Depending on the ownership type you chose, choose an accepted India document type to use for verification and type your document information.

> **Note**
>
> If you have a personal account and plan to use a driving license that is *not* issued by the Union of India, we recommend using a different personal document type for verification.

13. Choose the name you want to use for customer verification.

    The names from your billing and contact information will appear for selection if they are associated with an Indian address. Make sure the name you choose matches the name on the document type you plan to use for customer verification. If you need to make changes to the name associated with your billing or contact address, you can do so after you complete account sign up.
14. Provide your consent to submit the information for verification, and then choose **Continue**.

You will be notified of the customer verification result by email after you complete account signup. You can also check the status on the profile page in My AWS Settings. You must pass customer verification to access AWS services.

15. AWS will provision your first project.

To access AWS Settings, choose **Manage projects** from the AWS Management Console. In AWS Settings, you can invite collaborators, create new projects, and modify your billing and other settings.

Manage your customer verification information

In compliance with Indian Computer Emergency Response Team (CERT-In) regulations, AWS is required to collect and validate your identity information before granting you new or continued access to AWS services. Your identity must be verified using the name from the India billing or contact address you provided. During verification, AWS will check if the document number is valid and if the name you provide matches the name associated with the document you use for customer verification. The name that you choose between your contact or billing information must exactly match the name that appears on the document.

To update your billing name, address, and other contact information, see Use AWS Settings. If you edit any information you previously used for customer verification, such as the name or India-based address from your billing or contact information, you may need to update and re-submit your customer verification information.

Check your customer verification status

You can view your customer verification status at any time on the **Manage profile** page. If your verification status is **Verification required** or **Verification failed**, create or update your customer verification information and submit it for verification.

Create your customer verification information

To complete customer verification, you will need to provide information from an accepted India document. For a list of accepted document types, see the section called “Accepted India documents for customer verification”.

1. Open AWS Settings at https://settings.aws.com.

2. In the main navigation pane, choose **Manage profile**.
3. For **Customer verification**, choose **Create customer verification**.
4. Choose the name that exactly matches the name on the document you plan to use for customer verification. For example, if you plan to verify a business account using a Certificate of Incorporation, you must provide the business name that appears on the document.
5. Provide the remaining information requested on the page. Depending on the document type you chose, you may need to upload a copy of both the front and back of the document. If you upload an image file, make sure all the information in the document is visible and legible.
6. Choose **Submit**.

   You will be notified of the customer verification result and any next steps by email or on the AWS Health Dashboard.

Edit your customer verification information

You can edit your customer verification information, such as your primary purpose of account registration, your organization type, and the name, document type, document upload, or document information you want to use for verification.

If you edit the name or document type to use for customer verification, or update any document information, saving the changes will require your identity to be verified again.

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**.
3. For **Customer verification**, choose **Customer verification**.
4. Choose **Edit**, and then update the information you want to change.

   As you update the information, note the following guidance:

   - If you choose a different name, the name must exactly match the name on the document you plan to use for customer verification. For example, if you plan to verify a business account using a Certificate of Incorporation, you must provide the business name that appears on the document.
   - If you choose a different document type, you will need to upload a copy of the front and back (if applicable) of the document. All the information in the document upload should be visible and legible.

- If you have a personal account and plan to use a driving license that is *not* issued by the Union of India, we recommend using a different personal document type for verification.
- For a list of accepted document types, see the section called "Accepted India documents for customer verification".

5. Choose **Submit**.

   If your identity must be verified again due to the type of changes you saved, you will be notified of the customer verification result and any next steps by email. You can also view the results by returning to the **Customer verification** page or in the AWS Health Dashboard.

Accepted India documents for customer verification

The following document types issued by the Indian government are accepted for customer verification.

> **Note**
>
> The links shared below are subject to change by the government.

- **PAN Card** - Available in both digital and physical formats, the Permanent Account Number (PAN) card contains a unique alphanumeric identifier issued by the Income Tax Department of India to individuals, companies, and entities. A PAN consists of ten characters, including letters and numbers, in the format **AAAAA1111A**. To use this document for verification, you must also provide the date of birth (individual) or date of incorporation (business) that appears on the PAN document and upload the front side of the card. You can go to the Income Tax Department's official website to check the validity of your PAN.
- **Voter ID Card/EPIC** - The voter ID card, also known as the Electors Photo Identity Card (EPIC), contains a unique identification number issued by Election Commission of India to eligible voters in India. A voter ID/EPIC number consists of ten characters, including letters and numbers. You can go the official website of the Election Commission of India to check the validity of your voter ID. To use this document for verification, you must upload both the front and back side of the card.
- **Driving License** - If your driving license is *not* issued by the Union of India, we recommend using a different document type for verification. A driving license number consists of 12-16 characters, including letters, numbers, and a space or hyphen. To use this document for verification, you

must provide your date of birth and upload both the front and back side of the card. You can go to the Parivahan Sewa site of the Ministry of Road Transport and Highways to check the validity of your driving license.

- **Certificate of Incorporation** - A certificate of incorporation is a document issued by the Ministry of Corporate Affairs (MCA) that dates the registration of a business as a legal entity. The certificate is used to uniquely identify and track companies registered in India. Each certificate contains a Corporate Identification Number (CIN), which is a unique alphanumeric identifier consisting of 21 characters, including letters and numbers. To use this document for verification, you must upload the certificate of incorporation document. You can go to the Ministry of Corporate Affairs portal to check the validity of your CIN.

Different India document types are accepted for personal and business accounts:

- For personal accounts - PAN card, voter ID card/EPIC, and driving license.
- For business accounts - PAN card and certificate of incorporation.

Set up your India billing in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

If you sign up for AWS and choose India for your contact and billing address, your user agreement is with Amazon Web Services India Private Limited (AWS India), a local AWS seller in India. AWS India manages your billing, and your invoice total is listed in rupees instead of dollars. All tax information can only be created or modified by the project owner.

For additional tax or billing information, you can use the AWS Billing console for each project.

Use the **Plan & billing** section in the **Manage profile** page of AWS Settings to perform the following tasks:

- Adding or editing a Permanent Account Number
- Editing multiple Permanent Account Numbers
- Editing multiple Goods and Services Tax numbers

- Viewing a tax invoice

Adding or editing a Permanent Account Number

You can add your Permanent Account Number (PAN) to your account and edit it.

To add or edit a PAN

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**, and then choose **Plan & billing**.
3. On the **Tax Settings** navigation bar, choose **Edit**.
4. For **Permanent Account Number (PAN)**, enter your PAN, and then choose **Update**.

Editing multiple Permanent Account Numbers

You can edit multiple Permanent Account Numbers (PANs) in your account.

To edit multiple PAN numbers

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**, and then choose **Plan & billing**.
3. Under **Manage Tax Registration Numbers**, select the PAN numbers that you want to edit.
4. For **Manage Tax Registration**, choose **Edit**.
5. Update the fields that you want to change, and then choose **Update**.

Editing multiple Goods and Services Tax numbers

You can edit multiple Goods and Services Tax numbers (GSTs) in your account.

To edit multiple GST numbers

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**, and then choose **Plan & billing**.
3. Under **Manage Tax Registration Numbers**, select the GST numbers that you want to edit or choose **Edit all**.
4. For **Manage Tax Registration**, choose **Edit**.

5. Update the fields that you want to change and choose **Update**.

Viewing a tax invoice

You can view your tax invoices in AWS Settings.

To view a tax invoice

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**, and then choose **Plan & billing**.
3. Scroll down and choose the **Invoices** tab.
4. In the **Tax invoices** section, choose an invoice link that is mentioned under **Document ID**.

> **Note**
>
> The **Tax invoices** section only appears if there are tax invoices available.

Changing the seller of record to or from AWS India

You can change your AWS account seller of record (SOR) to another SOR by updating your billing or tax information. This does not apply if your AWS India account is operating in the legacy model where the functionality isn't available. When you edit your country or region in your billing or tax details, the SOR for your account will automatically change if the service provider of the new location is different from your current location.

If you're changing your account to or from AWS India, you do not need to update your default payment method if it is set to invoice or wire transfer. If your default payment method is credit or debit card, you are required to reconfirm your default payment card details. This is to ensure the payment method is stored separately from the transactions outside of AWS India. For more information, see Viewing eligible credit or debit cards for AWS India invoices.

> **Note**
>
> If you have an AWS India account with a credit or debit card as your default payment method and you have e-mandate set up for automatic payments, you must deactivate the e-mandate before you can edit your default payment preferences.

To change the SOR of your account, you can edit this information using the following options:

To change the SOR using billing and tax information

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**, and then choose **Plan & billing**.
3. In the **Default payment preferences** section, choose **Edit**.
4. Enter the new billing address.
5. In the **Payment currency** section, choose a default currency from the list of accepted currencies of your new SOR.
6. (If your default payment method is credit or debit card) Under the **Confirm payment method** section, reenter the details of your default payment card to confirm usage under the new service provider.

   To change default cards under the new SOR, add your card at a later time and change your default payment method setting.
7. Choose **Save changes**.

   After a few minutes, your changes including the updated SOR appears on the **Payment preferences** page.

To change the SOR using billing and tax information

To move your account to a different SOR, you must update the tax registration number (TRN) associated with the account and provide a new, valid TRN. You can also delete the TRN associated with the account. The new SOR of your account automatically resolves based on the country of your new tax address, and your current billing address, in order of availability and priority. When your account's SOR changes, you will receive a notification that your account is moving in or out of AWS India.

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Manage profile**, and then choose **Plan & billing**.
3. Select your account and choose **Manage tax registration**, then choose either **Edit**, **Edit all**, or **Delete TRN**.
4. (If your default payment method is credit or debit card) In the main navigation pane, choose **Manage profile**, and then choose **Plan & billing**.

5. Choose **Add payment method** to reenter the details of your default payment card, or add a new payment card.
6. After the card is saved, choose **Set as default**.

   The **Default payment preferences** section alerts you that your previous default payment method is not eligible for use in the new SOR. Refresh the page after you add your payment card for this alert to disappear.

Access AWS Support

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you sign up for our new AWS experience, there are two places where you can work with AWS Support.

Project support

If you are running into issues with your project, your team management, or issues related to billing, you can access Basic Support in AWS Settings. You can request additional projects for your organization or request quota increases related to billing. Since you're on Basic Support, you'll receive a response within 24 hours.

To create a support case

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Support**.
3. For **Project support**, choose **Create case**.
4. For **Subject**, enter the subject of your support case.
5. For **Description**, describe your issue. If you can, describe steps to reproduce the issue.
6. Choose **Create case**.

After you create a support case, you can view your case information to continue working with support.

To view your case information

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Support**.
3. Choose your support case. In this view you can see all communication about your case.
4. To send a reply to support, in **Reply**, enter your reply and then choose **Send reply**.

When your case is resolved, you can resolve it to remove it from AWS Settings.

To resolve your case

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Support**.
3. Choose **Resolve case**.

You won't be able to access the case details after it's been resolved.

Technical support

If you are having issues with your AWS resources, you need technical support. To access technical support, open a case in the AWS Support Center for the project that has the issue.

Support if all your projects and organization are closed

If all your projects and your organization are closed, contact support by filling out the Support Feedback form.

In the **Request information** section, under **How can we help you**, include that you are using our new AWS experience and that all your projects are closed.

Monitor your project

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

Monitor your project using AWS CloudTrail

Your project is integrated with AWS CloudTrail, a service that provides a record of actions taken by a team member or AWS service. CloudTrail captures all API calls for your project as events. The calls captured include calls from the console and API operations for all supported AWS services. Using the information collected by CloudTrail, you can determine the request that was made to your project, the person who made the request, when it was made, and additional details.

Use CloudTrail to see who modified your project

This tutorial is for users that are comfortable with the AWS CLI. If you share your project with multiple team members, you can use CloudTrail to view who modified any resources in your project. Every action taken by you or team members when they use AWS Settings or the AWS Management Console to access your project is logged by the `onBehalfOf` parameter. This parameter shows a user ID and an identity store ID. Together, these values define a builder ID that you've invited to access your project.

To find the `onBehalfOf` parameter and connect it to a team member, you'll need access to the AWS CLI and the AWS Management Console. If you don't have the AWS CLI configured, you can use AWS CloudShell to access a browser-based, pre-authenticated shell that you can launch directly from the AWS Management Console and run AWS CLI commands.

Step 1. To find an AWS CloudTrail Event

1. Sign in to the AWS Management Console and open the CloudTrail console at https://console.aws.amazon.com/cloudtrail/.
2. In the navigation pane, choose **Event history**. You see a filtered list of events, with the most recent events showing first. Choose an event.
3. In the event record, you'll see the following content:

```
"onBehalfOf": {
            "userId": "00000-e00f-000-000-0000000",
            "identityStoreArn":
 "arn:aws:identitystore::111122223333:identitystore/000000-0000-0000-0000-60000007"
}
```

This is the identity that defines the team member who created the CloudTrail event.

Step 2. To connect the onBehalfOf parameter to a team member

To connect the `onBehalfOf` parameter to a team member using the AWS CLI:

To use the AWS CLI in your terminal

1. Run the following command to sign in to the AWS CLI. You can find your Region using AWS Regions for your projects.

```
aws login --region project-region
```

2. Run the following command to get the ID for your identity store. The CloudTrail event shows the identity store ARN, but does not indicate the identity store ID. You always use the AWS Region `us-east-1` when finding the ID for your identity store.

```
aws sso-admin list-instances --region us-east-1
```

The output will look like the following:

```
{
    "Instances": [
        {
            "InstanceArn": "arn:aws:sso:::instance/ssoins-0000000000",
            "IdentityStoreId": "d-000000",
            "OwnerAccountId": "111122223333",
            "CreatedDate": "2025-10-26T23:10:58.529000+00:00",
            "Status": "ACTIVE",
            "PrimaryRegion": "us-east-1",
            "Regions": [
                {
                    "RegionName": "us-east-1",
```

```
                    "Status": "ACTIVE",
                    "AddedDate": "2025-10-26T23:10:58.529000+00:00",
                    "IsPrimaryRegion": true
                }
            ]
        }
    ]
}
```

You want to copy the identity store ID. In this case, the identity store ID is `d-000000`.

3. Run the following command to describe the team member who corresponds to the `onBehalfOf` parameter.

```
aws identitystore describe-user --user-id 00000-e00f-000-000-0000000 --region us-
east-1 --identity-store-id d-000000
```

The output will look like the following:

```
{
    "IdentityStoreId": "d-000000",
    "UserId": "00000-e00f-000-000-0000000",
    "UserName": "carlos_salazar",
    "Name": {
        "FamilyName": "Salazar",
        "GivenName": "Carlos"
    },
    "DisplayName": "Carlos",
    "Emails": [
        {
            "Value": "carlos_salazar@example.com",
            "Type": "work",
            "Primary": true
        }
    ],
    "UserStatus": "ENABLED",
    "CreatedAt": "2025-10-26T23:15:27.213000+00:00",
    "CreatedBy": "00000000000",
    "UpdatedAt": "2025-10-26T23:18:55.432000+00:00",
    "UpdatedBy": "0000000000"
}
```

Your team member's display name and emails are provided in this output.

To connect the `onBehalfOf` parameter to a team member using AWS CloudShell, log into the AWS Management Console and access CloudShell. For more information, see Getting started with AWS CloudShell. You will automatically have the correct IAM permissions to access CloudShell.

Close a project in AWS Settings

Warning

We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

If you no longer need your project, you can close it. After you close your project, you can reopen it within 90 days from the day you closed it. The time between the day you closed the project and when AWS permanently closes the project is the post-closure period.

What you need to know before closing your project

Before you close your project, consider the following:

- You cannot close your project if it is paused due to exceeding a spend limit. You need to turn off spend limits and then close the project. For more information, see Update a spend limit in AWS Settings.
- All team members will automatically lose access to this project and all the resources inside of it.
- Closing your project will serve as your notice of termination of the AWS Customer Agreement for this project.
- You don't need to delete resources in your project before closing it. However, we recommend you back up any resources or data that you want to keep. For instructions on how to back up a particular resource, see the appropriate AWS documentation for that service.
- You can reopen your project during the post-closure period. Charges for the services that remained in your project will restart if you reopen it. You also remain responsible for any unpaid invoices.
- You remain responsible for all outstanding fees and charges for the services consumed before the project closure. You will receive an AWS bill the following month after closing your project.

For example, if you closed your account on January 15, you will receive a bill at the beginning of February for usage incurred from January 1 through January 15.

- After a project has been closed, AWS will send daily emails for up to five days before we suspend the domain. After the domain has been suspended, and depending on the domain's registrar, we will either delete the domain within 30 days or release the domain to its registrar. For more information, see My AWS account is closed or permanently closed, and my domain is registered with Route 53. This information is written for AWS accounts, but the information is accurate for projects.
- AWS CloudTrail is a foundational security service. This means that trails created by users can continue to exist and deliver events even after a project is closed, unless a user explicitly deletes the trails in their project before closing it. For more information about how to request trail deletion after an AWS account has been closed, see AWS account closure and trails in the CloudTrail User Guide. This information is written for AWS accounts, but information is accurate for projects.

To close a project

To close a project

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose the project you want to access.
3. For **Close project**, choose **Close project**.
4. Confirm your choice and choose **Close project**.

The project page will still show your project for 90 days until it is permanently deleted.

Post-closure period

The post-closure period refers to the length of time between the day you closed your account and when AWS permanently closes your project. The post-closure period is 90 days. During the post-closure period, you can access your content and AWS services only by reopening your project. After the post-closure period, AWS permanently closes your project, and you can no longer reopen it. AWS will also delete content and resources in your account (except for CloudTrail trails). After a project has been permanently closed, its project ID can never be reused.

Reopening your project

Your project will permanently close in 90 days, after which you will not be able to reopen your project and AWS will delete the content remaining in your project. To reopen your project before it is permanently closed, (1) you must contact Support as soon as possible, and (2) we must receive full payment of any outstanding balance, including providing required information as specified on the invoice, within 60 days from the date of account closure.

> **Note**
>
> Charges for the services that remained in your project will restart if you reopen it.

Activate advanced AWS features

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

When you activate advanced features, you take complete control of the AWS Organization, management account, and delegated administrator account that are used to manage the projects you created and team members you invited. You will directly manage the security, governance, and team membership of your AWS environment. You have access to additional AWS services, multi-Region capabilities, and enhanced administrative and billing controls.

You must upgrade your account before you activate advanced features. For more information, see Upgrade your account in AWS Settings.

What happens when you activate advanced features

When you activate advanced features, the following changes will immediately be applied to your management account and the AWS Organization that contains your projects:

- You are now the administrator of the AWS organization that contains your projects.
- You manage your projects as member accounts in your organization. As the administrator of this organization, you can create member accounts in the organization.

- To give member accounts that you invite to join your organization an administrator role, you must create an OrganizationAccountAccessRole. For more information, see Creating the OrganizationAccountAccessRole in an invited member account.
- Your team members are referred to as workforce identities and they can still access your AWS accounts. You can also choose to customize the identity source of your workforce using IAM Identity Center. You can continue to use AWS Builder ID as the identity source, or you can change this to an external identity source. If you change your identity source from AWS Builder ID to another identity source, it is irreversible. You can't re-enable AWS Builder ID as an identity source in the future. For more information, see Change identity source from AWS Builder ID.
- You control AWS access for users and groups in your IAM Identity Center organization instance using Account access manager. You can use this feature to modify user access, while still allowing your users to sign in using `aws login` to receive their IAM role session credentials from the browser. For more information, see AWS Account Access Manager.
- AWS creates a member account called Identity Delegated Admin in your organization with a delegated administrator role. This role is used to perform most administrative tasks in the IAM Identity Center and account access manager. The delegated administrator role must be used in US East (N. Virginia). For more information, see Delegated administration.
- You can now configure Resource Control Policies (RCPs) and Service Control Policies (SCPs) to enforce custom guardrails in your organization. Previously, AWS managed RCPs and SCPs on your behalf. You can modify those policies after you activate advanced features. For more information, see Remove organization policies.

When you activate advanced features, the following changes are also applied to your AWS organization:

- If you have a spend limit for any of your AWS accounts, it is removed. You cannot create a spend limit if you activate advanced features. Instead, you have access to the entire suite of Billing and Cost Management Tools. This includes setting budgets, downloading cost explorer reports, and detecting unusual spend with AWS Cost Anomaly Detection. You can use these advanced tools to analyze, organize, plan, and optimize your costs.
- You can access all AWS services. The available services are listed in the section called “AWS services not supported for our new AWS experience”. You should plan your architecture and consult the service documentation before enabling these services. In addition, Agent Toolkit provides many skills to work with these services.

- You gain access to additional AWS Regions, opt-in Regions, and access to multi-Region capabilities. Multi-Region architecture improves resilience for your workloads. We recommend that you plan for how you will constrain and audit your regional footprint. It's important to ensure that resources you create are only present in the Regions you intend to use.
- You can turn off IAM role manager for each AWS account in your organization. IAM role manager is an optional account setting that automatically provisions roles, so you and your workforce don't need to set them up. For more information, see How to enable and disable role manager (console).
- You can use IAM Access Analyzer.

When you activate advanced features, the following changes are also applied to your user experience:

- Previously, the AWS Management Console showed a simplified view with a menu that you used to change between projects and access AWS Settings. The AWS Management Console expands to include additional features such as Region selector, additional services, and setting a color for account that you can access to visibly distinguish between them. For more information, see Console features.
- You can now also do the following:
  - Disable multi-session console support.
  - Designate your AWS account as HIPAA or SEC compliant.
  - Use AWS Marketplace.
  - Earn extra credits from AWS Activate.
  - Work with an AWS Training Partner.
  - Purchase an AWS Skill Builder Team subscription.
  - Enroll in an Enterprise Agreement with AWS.
  - Create a Professional Services contract.
  - Join AWS Partner Network.

How will AWS Settings change after I activate advanced features?

After you activate advanced features, AWS Settings will be available as long as you use AWS Builder ID as your identity source. AWS Settings will only be accessible from https://settings.aws.com. You won't be able to access it from the AWS Management Console. AWS

Settings will provide links to manage your account using your management account. With your management account, you can access the following AWS Management Console locations from AWS Settings. Only use the management account to perform administrative tasks, including:

- Access the Billing and Cost Management Console to view your billing information.
- Access the AWS Organization console to modify the SCPs and RCPs that govern your organization.
- Access the AWS Account Access Manager console to modify the fine grain access for your workforce identities.
- Access the IAM Identity Center to change your identity source.

You can also use AWS Settings to access your AWS accounts in your organization, and any project shared with you.

If you want to create custom roles and access them from AWS Settings in your accounts, you must set the session length to 12 hours.

If you have team member access to a project, you can view and access it in the same way as before you activated advanced features. Any projects shared with you are not impacted if you activate advanced features for your own management account and AWS organization.

Can I still create a project in AWS Settings after I activate advanced features?

You cannot create a project in your current AWS Organization after you activate advanced features. You must create a member account in your organization.

You can create a new managed AWS Organization with preconfigured defaults. To do this, create a new project in AWS Settings. You can then create additional projects and invite team members to collaborate. You can activate advanced features for this management account as well.

The new preconfigured environment that you access will have no connection to the previous AWS organization that was created when you first signed up for AWS.

How to prepare to activate advanced features

We recommend doing the following before you activate advanced features:

- Inform all your team members that you will be activating advanced features. Unless you modify any settings, your team members will have the same access to the projects, but the AWS Management Console changes from the simplified view to a view that shows Regions and provides access to more services. If you later change the identity source or add new SCPs or RCPs, inform your workforce since those changes can affect how they sign in and what they can access.
- View the spend for all your current projects and note if any projects contain costs that significantly fluctuate. Without a spend limit, you will be responsible for all costs incurred by your resources. We recommend that you delete or pause idle resources before you activate advanced features.
- Plan what changes you'll be making to your AWS organizations and research best practices. If you plan on creating resources in a different AWS Region, modify the service control policy that is applied for all users to restrict AWS Regions. You need to modify the statementID for `RegionFloor` to include any new Regions. For more information, see Service control policies for projects.
- If you followed the steps in Connect an AI coding tool, your AI tool has context only about the new AWS experience. This can cause problems as you develop using advanced features. We recommend the following:
  - Remove the `aws-starter-rules.md` file from the location you saved it. The rules file was saved depending on your agent. For more information, see Where to put the rules file.

How to activate advanced features

You can activate advanced features in AWS Settings. Activation cannot be reversed.

To activate advanced features

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Projects**.
3. For **Actions**, choose **Explore advanced features**.

   Because activation cannot be reversed, we'll provide more time to explore and learn about the activation process.
4. When you are ready to activate, do the following:

a. For **Team name**, enter a team name. This is the name of the IAM Identity Center instance used to store your workforce identities. Your workforce will see this name.

b. For **Choose management account email address**, choose an email address. This email address is used to access the management account and perform tasks. If you use a different email address, the AWS Builder ID email will still have access to the management account from AWS Settings, as long as you still use AWS Builder ID as your identity source.

c. Complete the verification process for the management account email address.

d. Choose **Review and complete activation**.

e. Confirm your choice and choose **activate**. If you do this, you cannot revert your account.

Congratulations, you've activated advanced AWS features!

After you activate advanced features

After you activated advanced features, you can do the following:

- Manage workforce members
- Change identity source from AWS Builder ID
- Remove organization policies
- Create an organizational unit
- Enable or disable AWS Regions in your account
- Create IAM roles

Manage workforce members

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

After you've activated advanced features, you can invite new workforce members or remove existing workforce members. This includes adding your existing workforce to new AWS accounts. The following information is only for AWS organizations that use AWS Builder ID as the identity

source. If you use an external identity provider (IdP), you manage your workforce in those identity providers. If you use Identity Center directory, use the documentation described in Manage your identity source.

To invite new workforce members

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Projects**.
3. For **Actions**, choose **Manage team**.

   This will open the IAM Identity Center.
4. On the **Users** page of the IAM Identity Center console, choose **Invite new team member**.
5. For **Email**, enter an email address or a list of email addresses. Separate the email addresses with commas (,) or semicolons (;).
6. Choose **Send invitation**.

After the workforce member accepts their invitation, you configure which AWS accounts they have access to in AWS Account Access Manager. For more information, see Manage access to AWS accounts.

To remove workforce members

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Projects**.
3. For **Actions**, choose **Manage team**.

   This will open the IAM Identity Center.
4. On the **Users** page of the IAM Identity Center console, select a team member, and then choose **Remove**.
5. Confirm your choice and choose **Remove**.

The team member will immediately lose all access to your AWS organization.

These steps show you how to manage your workforce members by first accessing AWS Settings. However, you can sign into the AWS Management Console with your delegate admin account and access the IAM Identity Center or the IAM console to manage your workforce members.

Change identity source from AWS Builder ID

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

After you've activated advanced features, you can change your identity source from AWS Builder ID. If you change your identity source from AWS Builder ID, you can never reuse AWS Builder ID as the identity source for your AWS organization.

To change your identity source from AWS Builder ID

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Projects**.
3. In **Manage this organization**, choose **Change identity source**.

   This will open the IAM Identity Center.
4. On the **Settings** page of the IAM Identity Center console, choose **Actions**, and then **Change identity source**.
5. For **Choose identity source**, AWS Builder ID will be selected. Choose a new source and then choose **Next**.
6. If you are changing to Active Directory, choose the available directory from the menu on the next page. If you are switching to an external identity provider, we recommend that you follow the steps in How to connect to an external identity provider.
7. Review and confirm your changes. You will need to select all checkboxes to confirm you understand the changes to your organization.
8. Confirm your choice and choose **Change identity source**.

These steps show you how to change your identity source by first accessing AWS Settings. However, you can sign into the AWS Management Console with your delegate admin account and access the IAM Identity Center to change your identity source.

Remove organization policies

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

After you've activated advanced features, you can modify the Resource Control Policies (RCPs) and Service Control Policies (SCPs) to change the governance of your AWS organization. Certain AWS managed organizational policies are removed when you activate advanced features. For more information, see Managed policies for your organization. The following information is only for AWS organizations that use AWS Builder ID as the identity source. If you use another source, follow the steps in Updating a policy.

To modify a service control policy (SCP)

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Projects**.
3. In **Manage this organization**, choose **Manage policies**.

   This will open the AWS Organizations console.
4. Choose the **Service control policy** page.
5. Choose the name of the policy that you want to update.
6. On the policy's detail page, choose **Edit policy**.
7. Make any or all of the following changes:

   - You can rename the policy by entering a new name in **Policy name**.
   - You can change the description by entering new text in **Policy description**.
   - You can edit the policy text by editing the policy in JSON format in the left pane. Alternatively, you can choose a statement in the editor on the right, and also alter its elements by using the controls. For more details about each control, see the Creating an SCP procedure earlier in this topic.
8. When you're finished, choose **Save changes**.

To modify a resource control policy (RCP)

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Projects**.
3. In **Manage this organization**, choose **Manage policies**.

   This will open the AWS Organizations console.
4. Choose the **Resource control policy** page.
5. Choose the name of the policy that you want to update.
6. On the policy's detail page, choose **Edit policy**.
7. Make any or all of the following changes:

   - You can rename the policy by entering a new name in **Policy name**.
   - You can change the description by entering new text in **Policy description**.
   - You can edit the policy text by editing the policy in JSON format in the left pane. Alternatively, you can choose a statement in the editor on the right, and also alter its elements by using the controls. For more details about each control, see the Creating an RCP procedure earlier in this topic.
8. When you're finished, choose **Save changes**.

These steps show you how to modify organization policies by AWS Settings. However, you can sign into the AWS Management Console with your management account and access the AWS Organizations console to modify these policies.

Close your account in AWS Settings

> **Warning**
>
> We're currently releasing our new experience to a limited number of customers. You might not be able to access this experience yet.

If you no longer need the organization and all the projects stored in it, you can close your projects, close your management account, and delete your AWS Builder ID.

To make sure you are no longer charged for any AWS resources you consume, you must close your projects and close the management account for your organization, but you don't have to close your AWS profile. Your AWS Builder ID associated with it is free.

After you close your management account, you can reopen it within 90 days from the day you closed it. The time between the day you closed the management account and when AWS permanently closes the management account is the post-closure period.

What you need to know before closing your management account

Before closing your management account, you should consider the following:

- When you close a project, consider the considerations in What you need to know before closing your project.
- You must close all projects before you can close your management account.
- You must close your management account before you delete your AWS Builder ID.
- After you close your management account, you can reopen it within 90 days from the day you closed it. The time between the day you closed the management account and when AWS permanently closes the management account is the post-closure period.
- You can reopen your account during the post-closure period.
- You can't use the same email address that was registered to your organization at the time of its closure as the primary email of another organization. If you want to use the same email address for a different organization while using Sign up for AWS (new), we recommend updating it before closure. For more information, see Change your email address in AWS Settings.
- If you've enabled multi-factor authentication (MFA) on your management account and configured an MFA device, MFA isn't removed automatically when you close the management account. If you choose to leave MFA turned on during the 90 days post-closure period, keep the MFA device active until the post-closure period has expired in case you need to access the account during that time. Note, the hardware TOTP token devices cannot be associated with another user after the permanent closure of your account. If you would like to use the hardware TOTP token with another user later, you have the option to deactivate the hardware MFA device before closing the management account.

Close a project

To close your projects, see Close a project in AWS Settings.

Close your management account

To close your management account

1. Open AWS Settings at https://settings.aws.com.
2. In the main navigation pane, choose **Project**.
3. For **Actions**, choose **Close management account**.
4. Confirm your choice and choose **Close management account**.

Delete your AWS Builder ID

Deleting your AWS Builder ID will result in the following:

- **Loss of access:** In addition to your project and organization, you can no longer access any AWS tools and services that you previously accessed through AWS Builder ID.
- **Content deletion:** Any remaining content associated solely with your AWS Builder ID will be deleted and you will no longer be able to access or recover your content from applications using your AWS Builder ID.
- **Personal information deletion:** Any personal information you provided in connection with the creation and administration of your AWS Builder ID will be deleted, except that AWS may retain personal information as required or permitted by law, such as records of your deletion request or data in a form that does not identify you.

You can find out more about how we handle your information in the AWS Privacy Notice. You can update your AWS communication preferences or unsubscribe by visiting the AWS Communications Preferences Center.

Existing logins remain unchanged – If you are using a login such as Google or Apple, deleting your AWS Builder ID does not delete anything related to your login account.

To delete your AWS Builder ID

1. Open AWS Settings at https://settings.aws.com.

   Even though you've deleted your management account, you can still sign in.
2. In the main navigation pane, choose **Manage profile**.
3. For **Profile info**, choose **Manage profile**.

You'll be redirected to your AWS Builder ID profile.

4. In the main navigation pane, choose **Privacy & data**.
5. For **Deleting AWS Builder ID**, choose **Delete your AWS Builder ID**.
6. Confirm your choice and choose **Delete**.

Sign up for AWS (advanced)

If you're new to AWS, the first step is to sign up for an AWS account. When you follow the information for Sign up for AWS (advanced), AWS creates an account where you have complete control over account configuration and where your content is stored. Use this option if you have regulated workloads such as HIPAA, FedRAMP.

To compare this sign up with Sign up for AWS (new), see Compare sign-up options.

To sign up for an AWS account, you'll need to provide the following information:

- **Root user email address** – The email address is used as the sign-in name for the root user and is required for account recovery. You must be able to receive email messages that are sent to this address. Before you can perform certain tasks, you must verify that you have access to email sent to this address.
- **AWS account name** – The name of the account appears in several places, such as on your invoice, and in consoles such as the Billing and Cost Management dashboard and the AWS Organizations console. We recommend that you use a standard way to name your accounts so that you can give your accounts names that are easy to recognize. For company accounts, consider using a naming standard such as *organization-purpose-environment* (for example, *AnyCompany-audit-prod*). For personal accounts, consider using a naming standard such as *first name-last name-purpose* (for example, *paulo-santos-testaccount*).
- **Address** – If your contact and billing address is in India, the user agreement for your account is with Amazon Web Services India Private Limited (AWS India), a local AWS seller in India. You must provide your CVV as part of the verification process. You might also have to enter a one-time password, depending on your bank. AWS India charges your payment method 2 INR as part of the verification process. AWS India refunds the 2 INR after verification is complete.
- **Phone number** – This number is used for identity verification purposes and to confirm the ownership of your account. You must be able to receive calls and SMS messages at this phone number.

> **Important**
>
> If this account is for a business, use a corporate phone number so that your company can retain access to the AWS account even when an employee changes positions or leaves the company.

Step 1: Create your account

These instructions are for creating an AWS account outside of India. For creating an account in India, see Create an AWS account with AWS India. For creating an account that's part of an organization managed by AWS Organizations, see Creating a member account in an organization in the *AWS Organizations User Guide*.

AWS Management Console

To create an AWS account

1. Open the Sign up for AWS page.
2. Enter the root user email address and AWS account name, and then choose **Verify email address**. This sends a verification code to your email address.

   > **Important**
   >
   > If this account is for a business, use a secure corporate distribution list (for example, `it.admins@example.com`) so that your company can retain access to the AWS account even when an employee changes positions or leaves the company. Because the email address can be used to reset the account's root user credentials, protect access to this distribution list or address.

3. Enter your verification code, and then choose **Verify**.
4. Enter a strong password for your root user, confirm it, and then choose **Continue**. AWS requires that your password meet the following conditions:

   - It must have a minimum of 8 characters and a maximum of 128 characters.
   - It must include a minimum of three of the following mix of character types: uppercase, lowercase, numbers, and ! @ # $ % ^ & * () <> [] {} | _+-= symbols.
   - It must not be identical to your AWS account name or email address.

5. Choose your account plan. For more information, see Free Tier plans.
6. Enter your contact information, then read and accept the AWS Customer Agreement. Make sure you understand the terms before accepting.

> **Important**
>
> If this account is for a business, it's a best practice to enter a company phone number rather than a number for a personal phone. Configuring the account's root user with an individual email address or a personal phone number can make your account insecure.

7. Enter your billing information. If you want to use a different billing address for your AWS billing information, choose **Use a new address**.

   You can't proceed with the sign-up process until you add a valid payment method.

8. You might need to confirm your identity:

   a. For **Country or region code**, enter a phone number that can be reached in the next few minutes.

   b. For **Phone number**, enter a phone number that can be reached in the next few minutes.

   c. Choose **Send SMS**.

   d. When the automated system contacts you, enter the code you receive and then choose **Continue**.

9. Choose one of the available AWS Support plans. For a description of the available Support plans and their benefits, see Compare Support plans.

10. Choose **Complete sign up**. A confirmation page appears that indicates that your account is being activated.

11. Check your email and spam folder for an email message that confirms your account was activated. Activation usually takes a few minutes but can sometimes take up to 24 hours.

12. After you receive the activation message, you can sign-in to the AWS Management Console to start using AWS services. For general information about how to manage your account settings, see Configure your AWS account.

For multiple AWS accounts managed through AWS Organizations, assign administrative access to an administrative user in IAM Identity Center. For instructions, see Set up AWS account access for an IAM Identity Center administrative user in the *IAM Identity Center User Guide*.

AWS CLI & SDKs

You can create member accounts in an organization that is managed by AWS Organizations by running the CreateAccount operation while signed in to the organization's management account.

You can't create a standalone AWS account outside of an organization by using an AWS Command Line Interface (AWS CLI) or AWS API operation.

Step 2: (Recommended) Install the AWS CLI

Install the AWS CLI by following the instructions at Installing the AWS CLI. You need version 2.32.0 or later.

You can use the AWS CLI to have an agent manage AWS resources on your behalf.

After you install the AWS CLI, use the following command to sign in programmatically:

```
aws login
```

This automatically rotates your credentials every 15 minutes, keeping your session valid for up to 12 hours without manual intervention.

Use the following command to verify your credentials are working:

```
aws sts get-caller-identity
```

For more information about accessing your AWS account, see Accessing your AWS account.

Step 3: (Recommended) Set up the AWS MCP Server

The AWS MCP Server is a managed server that gives agents access to AWS through the Model Context Protocol (MCP). Agents can search AWS documentation and retrieve service information without authentication. To execute AWS API calls, run Python scripts in a sandboxed environment, or follow curated skills, agents authenticate through your existing IAM credentials. For more information, see What is the AWS Agent Toolkit?

After you complete the sign-up process, see the following topics:

- Related AWS services

- Using the AWS account root user
- Accessing your AWS account
- Plan your AWS account governance structure
- Configure your AWS account
- Troubleshoot your AWS account
- Manage accounts in India
- Close an AWS account

Related AWS services

Depending on how you sign up for AWS, you either have access to AWS accounts or projects. Projects contain AWS accounts and the settings for sharing with other collaborators. For more information, see Compare sign-up options. In this section, we explain how AWS accounts that you create using Sign up for AWS (advanced) work seamlessly with the following services:

- **IAM**

  Your AWS account is closely integrated with AWS Identity and Access Management (IAM). You can use IAM with your account to ensure that other people who work in your account have as much access as they need to get their jobs done. You also use IAM to control access to all of your AWS resources, not only account specific information. It's important that you familiarize yourself with the major concepts and best practices of IAM before you get too far along with setting up the structure of your AWS account. For more information, see Security best practices in IAM in the *IAM User Guide*.

- **AWS Organizations**

  If your company is large or likely to grow, you might want to set up multiple AWS accounts that reflect your company's specific structure. AWS Organizations provides the underlying infrastructure and capabilities for you to build and manage your multi-account environments. You can combine your existing accounts into an organization that enables you to manage the accounts centrally. You can create accounts that automatically are a part of your organization, and you can invite other accounts to join your organization. You also can attach policies that affect some or all of your accounts. For more information, see When to use AWS Organizations.

  The following account-related API actions are part of the AWS Organizations API, not the AWS Account Management API. We include them here because customers commonly look for

these operations in the Account Management documentation, but they are defined in the AWS Organizations namespace.

- CreateAccount
- CreateGovCloudAccount
- DescribeAccount

- **AWS Control Tower**

  AWS Control Tower provides a simplified way to set up and govern a secure, multi-account AWS environment. AWS Control Tower automates the creation of your multi-account environment using AWS Organizations, instantiating a set of initial accounts and with some default guardrails and configurations for the environment. You can use AWS Control Tower to provision new AWS accounts in a few steps while ensuring that the accounts conform to your organizational policies. For more information, see When to use AWS Control Tower.

Using the AWS account root user

Depending on how you sign up for AWS, you either have access to AWS accounts or projects. Projects contain AWS accounts and the settings for sharing with other collaborators. For more information, see Compare sign-up options. In this section, we explain how to use the root user in an AWS account that you create using Sign up for AWS (advanced).

When you create an AWS account, you begin with one sign-in identity called the AWS account *root user* that has complete access to all AWS services and resources. We strongly recommend that you don't use the root user for everyday tasks. For tasks that require root user credentials, see Tasks that require root user credentials in the *IAM User Guide*.

To avoid using the root user for everyday tasks, learn how to set up an administrative user in AWS IAM Identity Center. For additional root user security recommendations, see Root user best practices for your AWS account.

> **Important**
>
> Anyone who has root user credentials for your AWS account has unrestricted access to all the resources in your account, including billing information.

You can change, or reset the root user password, and create, or delete access keys (access key IDs and secret access keys) for your root user. For help signing in using your root user, see Sign in to the AWS Management Console as the root user in the *AWS Sign-In User Guide*.

Accessing your AWS account

This information is about accessing your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

You can access your AWS account in any of the following ways:

AWS Management Console

The AWS Management Console is a browser-based interface that you can use to manage your AWS account settings and your AWS resources.

AWS Command Line Tools

With the AWS command line tools, you can issue commands at your system's command line to perform AWS account and AWS tasks. Working with the command line can be faster and more convenient than using the console. The command line tools also are useful if you want to build scripts that perform AWS tasks. AWS provides two sets of command line tools:

- AWS Command Line Interface (AWS CLI). For information about installing and using the AWS CLI, see the AWS Command Line Interface User Guide.
- AWS Tools for Windows PowerShell. For information about installing and using the Tools for Windows PowerShell, see the AWS Tools for PowerShell User Guide.

AWS SDKs

The AWS SDKs consist of libraries and sample code for various programming languages and platforms (for example, Java, Python, Ruby, .NET, iOS, and Android). The SDKs take care of tasks such as cryptographically signing requests, managing errors, and retrying requests automatically. For more information about the AWS SDKs, including how to download and install them, see Tools for Amazon Web Services.

AWS Account Management HTTPS Query API

The AWS Account Management HTTPS Query API gives you programmatic access to your AWS account and AWS. The HTTPS Query API lets you issue HTTPS requests directly to the service.

When you use the HTTPS API, you must include code to digitally sign requests using your credentials. For more information, see Calling the API by making HTTP Query requests.

Plan your AWS account governance structure

This information is about planning your AWS account governance structure if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

Although you might have started your AWS journey with a single account, AWS recommends that you set up multiple accounts as your workloads grow in size and complexity. Whether you are a medium business or a large enterprise, you'll want to create a governance structure plan that will ensure your data and your workload needs are met.

This section covers the benefits and governance services available in AWS to help enable a multi-account governance structure.

Topics

Benefits of using multiple AWS accounts

This information is about the benefits of using multiple AWS accounts if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

AWS accounts form the foundational security boundary in the AWS Cloud. They serve as a container for resources, providing a critical layer of isolation that is essential for creating a secure, well-governed environment. For more information, see What is an AWS account?.

Separating your resources into separate AWS accounts helps you to support the following principles in your cloud environment:

- **Security control** – Different applications can have different security profiles, requiring different control policies and mechanisms around them. For example, it's far easier to talk to an auditor and be able to point to a single AWS account that hosts all elements of your workload that are subject to Payment Card Industry (PCI) Security Standards.
- **Isolation** – An AWS account is a unit of security protection. Potential risks and security threats should be contained within an AWS account without affecting others. There could be different security needs due to different teams or different security profiles.
- **Many teams** – Different teams have their different responsibilities and resource needs. You can prevent teams from interfering with each other by moving them to separate AWS accounts.
- **Data isolation** – In addition to isolating the teams, it's important to isolate the data stores to an account. This can help limit the number of people that can access and manage that data store. This helps contain exposure to highly private data and therefore can help in compliance with the European Union's General Data Protection Regulation (GDPR).
- **Business process** – Different business units or products may have completely different purposes and processes. With multiple AWS accounts, you can support a business unit's specific needs.
- **Billing** – An account is the only true way to separate items at a billing level. Multiple accounts help separate items at a billing level across business units, functional teams, or individual users. You can still get all of your bills consolidated to a single payer (using AWS Organizations and consolidated billing) while having line items separated by AWS account.
- **Quota allocation** – AWS service quotas are enforced separately for each AWS account. Separating workloads into different AWS accounts prevents them from consuming quotas for each other.

All of the recommendations and procedures described in this document are in compliance with the AWS Well-Architected Framework. This framework is intended to help you design a flexible, resilient, and scalable cloud infrastructure. Even when you are starting small, we recommend that you proceed in compliance with this guidance in the framework. Doing so can help you scale your environment securely and without impacting your ongoing operations as you grow.

Managing multiple AWS accounts

Before you start adding multiple accounts, you'll want to develop a plan to manage them. For that, we recommend that you use AWS Organizations, which is a free AWS service to manage all of the AWS accounts in your organization.

AWS also offers AWS Control Tower, which adds layers of AWS managed automation to Organizations and automatically integrates it with other AWS services like AWS CloudTrail, AWS Config, Amazon CloudWatch, AWS Service Catalog, and others. These services can incur additional costs. For more information, see AWS Control Tower pricing.

See also

- When to use AWS Organizations
- When to use AWS Control Tower

When to use AWS Organizations

This information is about when to use AWS Organizations if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

AWS Organizations is an AWS service that you can use to manage your AWS accounts as a group. This provides features like consolidated billing, where all of your accounts' bills are grouped together and handled by a single payer. You can also centrally manage the security of your organization by using policy based controls. For more information about AWS Organizations, see the AWS Organizations User Guide.

Trusted access

When you use AWS Organizations to manage your accounts as a group, most administrative tasks for the organization can be performed by only the organization's *management account*. By default, this includes only operations related to managing the organization itself. You can extend this additional functionality to other AWS services by enabling *trusted access* between Organizations and that service. Trusted access grants permissions to the specified AWS service to access information about the organization and the accounts it contains. When you enable trusted access for Account Management, the Account Management service grants Organizations and its management account permissions to access the metadata, such as the primary or alternate contact information, for all of the organization's member accounts.

For more information, see Enable trusted access for AWS Account Management.

Delegated admin

After you enable trusted access, you can also choose to designate one of your member accounts as a *delegated admin* account for AWS Account Management. This allows the delegated admin

account to perform the same Account Management metadata management tasks for the member accounts in your organization that previously only the management account could do. The delegated admin account can access only the management tasks for the Account Management service. The delegated admin account doesn't have all of the administrative access to the organization that the management account has.

For more information, see Enable a delegated admin account for AWS Account Management.

Service control policies

When your AWS account is part of an organization that is managed by AWS Organizations, then the administrator of the organization can apply service control policies (SCPs) that can limit what the principals in member accounts can do. An SCP never grants permissions; instead, it is a filter that limits what permissions can be used by the member account. A user or role (*a principal*) in a member account can perform only those operations that are in the intersection of what is allowed by the SCPs that apply to the account and the IAM permission policies attached to the principal. For example, you can use SCPs to prevent any principal in an account from modifying their own account's alternate contacts.

For example SCPs that apply to AWS accounts, see Restrict access using AWS Organizations service control policies.

Enable trusted access for AWS Account Management

These instructions are for how to enable trusted access for AWS Account Management if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

Enabling trusted access for AWS Account Management allows the administrator of the management account to modify the information and metadata (for example, primary or alternate contact details) specific to each member account in AWS Organizations. For more information, see AWS Account Management and AWS Organizations in the *AWS Organizations User Guide*. For general information about how trusted access works, see Using AWS Organizations with other AWS services.

After trusted access has been enabled, you can use the `accountID` parameter in those Account Management API operations that support it. You can use this parameter successfully only if you call the operation using credentials from the management account, or from the delegated admin account for your organization if you enable one. For more information, see Enable a delegated admin account for AWS Account Management.

Use the following procedure to enable trusted access for Account Management in your organization.

Minimum permissions

To perform these tasks, you must meet the following requirements:

- You can perform this only from the organization's management account.
- Your organization must have all features enabled.

AWS Management Console

To enable trusted access for AWS Account Management

1. Sign in to the AWS Organizations console. You must sign in as an IAM user, assume an IAM role, or sign in as the root user (not recommended) in the organization's management account.
2. Choose **Services** in the navigation pane.
3. Choose **AWS Account Management** in the list of services.
4. Choose **Enable trusted access**.
5. In the **Enable trusted access for AWS Account Management** dialog box, type **enable** to confirm it, and then choose **Enable trusted access**.

AWS CLI & SDKs

To enable trusted access for AWS Account Management

After running the following command, you can use credentials from the organization's management account to call Account Management API operations that use the `--accountId` parameter to reference member accounts in an organization.

- AWS CLI: enable-aws-service-access

  The following example enables trusted access for AWS Account Management in the calling account's organization.

```
$ aws organizations enable-aws-service-access \
```

```
    --service-principal account.amazonaws.com
```

This command produces no output if it's successful.

Enable a delegated admin account for AWS Account Management

These instructions are for how to enable a delegated admin account for AWS Account Management if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

You can register one delegated admin account per organization for the AWS Account Management service. After you register this account, users and roles in that account can call the AWS CLI and AWS SDK operations in the `account` namespace that can work in the Organizations mode by supporting an optional `AccountId` parameter. This enables the delegated admin to call AWS Account Management API operations for other member accounts in AWS Organizations.

To register a member account in your organization as a delegated admin account, use the following procedure.

AWS CLI & SDKs

To register a delegated admin account for the Account Management service

You can use the following commands to enable a delegated admin for the Account Management service.

Minimum permissions

To perform these tasks, you must meet the following requirements:

- You can perform this only from the organization's management account.
- Your organization must have all features enabled.
- You must have enabled trusted access for Account Management in your organization.

You must specify the following service principal:

```
account.amazonaws.com
```

- AWS CLI: register-delegated-administrator

  The following example registers a member account of the organization as a delegated admin for the Account Management service.

  ```
  $ aws organizations register-delegated-administrator \
      --account-id 123456789012 \
      --service-principal account.amazonaws.com
  ```

  This command produces no output if it's successful.

  After you run this command, you can use credentials from account 123456789012 to call Account Management AWS CLI and SDK API operations that use the `--account-id` parameter to reference member accounts in an organization.

AWS Management Console

This task isn't supported in the AWS Account Management management console. You can perform this task only by using the AWS CLI or an API operation from one of the AWS SDKs.

Restrict access using AWS Organizations service control policies

This information is about restricting access using AWS Organizations service control policies if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

This topic presents examples that show how you can use service control policies (SCPs) in AWS Organizations to restrict what the users and roles in the accounts in your organization can do. For more information about service control policies, see the following topics in the *AWS Organizations User Guide*:

- Creating SCPs
- Attaching SCPs to OUs and accounts
- Strategies for SCPs
- SCP policy syntax

Example 1: Prevent accounts from modifying their own alternate contacts

The following example denies the `PutAlternateContact` and `DeleteAlternateContact` API operations from being called by any member account in standalone account mode. This prevents any principal in the affected accounts from changing their own alternate contacts.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Deny",
            "Action": [
                "account:PutAlternateContact",
                "account:DeleteAlternateContact"
            ],
            "Resource": [ "arn:aws:account::*:account" ]
        }
    ]
}
```

Example 2: Prevent any member account from modifying alternate contacts for any other member account in the organization

The following example generalizes the `Resource` element to "*", which means that it applies to both standalone mode requests and organizations mode requests. This means that even the delegated admin account for Account Management, if the SCP applies to it, is blocked from changing any alternate contact for any account in the organization.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "Statement1",
            "Effect": "Deny",
```

```
            "Action": [
                "account:PutAlternateContact",
                "account:DeleteAlternateContact"
            ],
            "Resource": [ "*" ]
        }
    ]
}
```

Example 3: Prevent a member account in an OU from modifying its own alternate contacts

The following example SCP includes a condition that compares the account's organization path to a list of two OUs. This results in blocking a principal in any account in the specified OUs from modifying their own alternate contacts.

When to use AWS Control Tower

This information is about when to use AWS Control Tower if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

AWS Organizations is the foundational service that enables you to centrally manage and secure your entire AWS environment. A crucial component of this AWS Organizations-centric approach is AWS Control Tower. AWS Control Tower acts as a management console within Organizations, providing a streamlined way to set up and govern a secure, multi-account AWS environment by applying prescriptive best practices.

This security best practices approach provided by AWS Control Tower extends the core capabilities of AWS Organizations. AWS Control Tower applies a set of preventive and detective guardrails to help ensure your organization and accounts remain aligned with recommended security and compliance standards.

By establishing a well-architected AWS Organizations structure with AWS Control Tower, you can quickly deploy a scalable, secure, and compliant AWS environment. This centralized approach to cloud management and governance is essential for enterprises looking to harness the full power of the AWS Cloud while maintaining the highest standards of security and compliance.

For more information, see What is AWS Control Tower? in the *AWS Control Tower User Guide*.

Understanding API modes of operation

This information is about understanding API modes of operation if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

The API operations that work with an AWS account's attributes always work in one of two modes of operation:

- **Standalone context** – this mode is used when a user or role in an account accesses or changes an account attribute in the ***same account***. The standalone context mode is automatically used when you ***don't*** include the `AccountId` parameter when you call one of the Account Management AWS CLI or AWS SDK operations.
- **Organizations context** – this mode is used when a user or role in one account in an organization accesses or changes an account attribute in a different member account in the same organization. The organizations context mode is automatically used when you ***do*** include the `AccountId` parameter when you call one of the Account Management AWS CLI or AWS SDK operations. You can call the operations in this mode from only the management account of the organization, or the delegated admin account for Account Management.

The AWS CLI and AWS SDK operations can work in either standalone or organizations context.

- If you ***don't*** include the `AccountId` parameter, then the operation runs in the standalone context and automatically applies the request to the account you used to make the request. This is true whether or not the account is a member of an organization.
- If you do include the `AccountId` parameter, then the operation runs in the organizations context, and the operation works on the specified Organizations account.
  - If the account calling the operation is the management account or the delegated admin account for the Account Management service, then you can specify any member account of that organization in the `AccountId` parameter to update the specified account.
  - The only account in an organization that can call one of the alternate contact operations and specify its own account number in the `AccountId` parameter is the account specified as the delegated admin account for the Account Management service. Any other account, including the management account, receives an `AccessDenied` exception.

- If you run an operation in standalone mode, then you must be permitted to run the operation with an IAM policy that includes a `Resource` element of either `"*"` to allow all resources, or an ARN that uses the syntax for a standalone account.
- If you run an operation in organizations mode, then you must be permitted to run the operation with an IAM policy that includes a `Resource` element of either `"*"` to allow all resources, or an ARN that uses the syntax for a member account in an organization.

Granting permissions to update account attributes

As with most AWS operations, you grant permissions to add, update, or delete account attributes for AWS accounts by using IAM permission policies. When you attach an IAM permission policy to an IAM principal (either a user or role), you specify which actions that principal can perform on which resources, and under what conditions.

The following are some Account Management specific considerations for creating a permissions policy.

Amazon Resource Name format for AWS accounts

- The Amazon Resource Name (ARN) for an AWS account that you can include in the `resource` element of a policy statement is constructed differently based on whether the account you want to reference is a standalone account or an account that is in an organization. See the previous section on Understanding API modes of operation.
  - An account ARN for a standalone account:

    ```
    arn:aws:account::{AccountId}:account
    ```

    You must use this format when you run an account attributes operation in standalone mode by not including the `AccountID` parameter.
  - An account ARN for a member account in an organization:

    ```
    arn:aws:account::{ManagementAccountId}:account/o-{OrganizationId}/{AccountId}
    ```

    You must use this format when you run an account attributes operation in organizations mode by including the `AccountID` parameter.

Context keys for IAM policies

The Account Management service also provides several Account Management service-specific condition keys that provide fine-grained control over the permissions you grant.

`account:AccountResourceOrgPaths`

The context key `account:AccountResourceOrgPaths` lets you specify a path through your organization's hierarchy to a specific organizational unit (OU). Only member accounts that are contained by that OU match the condition. The following example snippet restricts the policy to apply to only accounts that are in either of two specified OUs.

Because `account:AccountResourceOrgPaths` is a multi-valued string type, you must use the `ForAnyValue` or `ForAllValues` multi-value string operators. Also, note that the prefix on the condition key is `account`, even though you are referencing paths to OUs in an organization.

```
"Condition": {
    "ForAnyValue:StringLike": {
        "account:AccountResourceOrgPaths": [
            "o-aa111bb222/r-a1b2/ou-a1b2-f6g7h111/*",
            "o-aa111bb222/r-a1b2/ou-a1b2-f6g7h222/*"
        ]
    }
}
```

`account:AccountResourceOrgTags`

The context key `account:AccountResourceOrgTags` lets you reference the tags that can be attached to an account in an organization. A tag is a key/value string pair that you can use to categorize and label the resources in your account. For more information about tagging, see Tag Editor in the *AWS Resource Groups User Guide*. For information about using tags as part of an attribute-based access control strategy, see What is ABAC for AWS in the *IAM User Guide*. The following example snippet restricts the policy to apply to only accounts in an organization that have the tag with the key `project` and a value of either `blue` or `red`.

Because `account:AccountResourceOrgTags` is a multi-valued string type, you must use the `ForAnyValue` or `ForAllValues` multi-value string operators. Also, note that the prefix on the condition key is `account`, even though you are referencing the tags on an organization's member account.

```
"Condition": {
```

```
    "ForAnyValue:StringLike": {
        "account:AccountResourceOrgTags/project": [
            "blue",
            "red"
        ]
    }
}
```

Note

You can attach tags to only an account in an organization. You can't attach tags to a standalone AWS account.

Configure your AWS account

These instructions are for how to configure your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

This section includes topics that describe how to manage your AWS account.

Note

If your AWS account was created in India by using Amazon Web Services India Private Limited (AWS India), there are additional considerations. For more information, see Manage accounts in India.

Topics

- Create an AWS account alias
- Enable or disable AWS Regions in your account
- Update billing for your AWS account
- Update the root user email address
- Update root user password
- Update your AWS account name
- Update the alternate contacts for your AWS account

- Update the primary contact for your AWS account
- View AWS account identifiers

Create an AWS account alias

These instructions are for how to create an AWS account alias if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

If you want the URL for your IAM users to contain your company name (or another easy-to-remember identifier) instead of the AWS account ID, you can create an *account alias*.

To learn how to create or update an account alias, see Using an alias for your AWS account ID in the *IAM User Guide*.

Enable or disable AWS Regions in your account

This information is about enabling or disabling AWS Regions in your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

An *AWS Region* is a physical location in the world where AWS has multiple Availability Zones. Availability Zones consist of one or more discrete AWS data centers, each with redundant power, networking, and connectivity, housed in separate facilities. This means that each AWS Region is physically isolated and independent of the other Regions. Regions provide fault tolerance, stability, and resilience, and can also reduce latency. Running workloads in an AWS Region closer to end users can improve performance and lower latency. For a map of available and upcoming Regions, see Regions and Availability Zones. To learn more about AWS Regions and resiliency architecture for your workloads, visit AWS multi-Region fundamentals.

AWS Regions broadly fall into two categories of availability for accounts:

- **Default Regions** – Regions launched before March 20, 2019 are enabled by default. You can create and manage resources in these default Regions immediately after your account activation. Default Regions cannot be enabled or disabled.
- **Opt-in Regions** – Regions launched after March 20, 2019 are disabled by default and referred to as opt-in Regions. Disabled opt-in Regions are not shown in the Console Navigation bar, and you cannot use these Regions to create workloads until they are enabled. To use these opt-in

Regions, you must first enable them in your AWS account. After enabling an opt-in Region, you can select that Region in the navigation bar and create and manage resources in that Region. To enable an opt-in Region for your standalone accounts, see Enable or disable a Region for standalone accounts and to enable an opt-in Region for your member accounts, see Enable or disable a Region in your organization.

When you sign up for an AWS account, AWS recommends an opt-in Region for you based on your contact address country. Customers in a country with an AWS opt-in Region see a recommendation on the Contact Information page to enable the opt-in Region in that country. Customers in a country with both an opt-in Region and a default Region, like India, Australia, or Canada, see a recommendation to select the opt-in Region if the opt-in Region is closer to them than the default Region. After an account is activated, you can enable other AWS opt-in Regions in your account or choose to disable the opt-in Region you enabled during sign-up.

When you create an AWS account, your IAM data and credentials are automatically configured to work across all default Regions, allowing the root user and IAM identities with appropriate permissions to access AWS services in these Regions using their existing credentials. AWS opt-in Regions are disabled by default, and IAM data and credentials are not initially available in these Regions, which prevents access to AWS services in that Region. When you choose to enable an opt-in Region, AWS propagates your IAM data and credentials to that Region. Once the propagation is complete and the opt-in Region is enabled, the root user and IAM identities can then access AWS services in the newly enabled opt-in Region using the same IAM credentials they use in default Regions.

When you disable an opt-in Region, your IAM credentials are deactivated and you lose IAM access to the resources in that opt-in Region. Disabling an opt-in Region does not delete the resources in that Region and charges for resources (if any) in that disabled opt-in Region continue to accrue at the standard rate.

Important

Disabling a Region disables IAM access to resources in the Region. This does not delete the resources in question, which continue to incur charges. Remove any remaining resources prior to disabling a Region.

AWS groups Regions into partitions. Every Region is in exactly one partition, and each partition has one or more Regions. Partitions have independent instances of AWS Identity and Access

Management (IAM) and provide a hard boundary between Regions in different partitions. AWS commercial Regions are in the `aws` partition, Regions in China are in the `aws-cn` partition, and AWS GovCloud (US) Regions are in the `aws-us-gov` partition. Depending on the partition where you created your AWS account, you can use AWS Regions within that partition.

- An account in `aws` partition provides you access to multiple Regions in the commercial partition so that you can launch AWS resources in locations that meet your requirements. For example, you might want to launch Amazon EC2 instances in Europe to be closer to your European customers or to meet legal requirements.
- An account in `aws-us-gov` partition provides you access to the AWS GovCloud (US-West) Region and the AWS GovCloud (US-East) Region. For more information, see AWS GovCloud (US).
- An account in `aws-cn` partition provides you access to the Beijing and Ningxia Regions only. For more information, see Amazon Web Services in China.

Topics

Regional availability reference

The following tables list AWS Regions by availability type. Default Regions are enabled automatically and cannot be disabled, while opt-in Regions must be manually enabled before you can use them:

Opt-in Regions

The following Regions are opt-in Regions that must be enabled before you can use them:

Name	Code	Status
Africa (Cape Town)	`af-south-1`	GA
Asia Pacific (Hong Kong)	`ap-east-1`	GA

Name	Code	Status
Asia Pacific (Taipei)	ap-east-2	GA
Asia Pacific (Hyderabad)	ap-south-2	GA
Asia Pacific (Jakarta)	ap-southeast-3	GA
Asia Pacific (Melbourne)	ap-southeast-4	GA
Asia Pacific (Malaysia)	ap-southeast-5	GA
Asia Pacific (New Zealand)	ap-southeast-6	GA
Asia Pacific (Thailand)	ap-southeast-7	GA
Canada West (Calgary)	ca-west-1	GA
Europe (Zurich)	eu-central-2	GA
Europe (Milan)	eu-south-1	GA
Europe (Spain)	eu-south-2	GA
Israel (Tel Aviv)	il-central-1	GA
Middle East (UAE)	me-central-1	GA
Middle East (Bahrain)	me-south-1	GA
Mexico (Central)	mx-central-1	GA

Default Regions

The following Regions are enabled by default and cannot be disabled:

Name	Code
Asia Pacific (Tokyo)	ap-northeast-1

Name	Code
Asia Pacific (Seoul)	ap-northeast-2
Asia Pacific (Osaka)	ap-northeast-3
Asia Pacific (Mumbai)	ap-south-1
Asia Pacific (Singapore)	ap-southeast-1
Asia Pacific (Sydney)	ap-southeast-2
Canada (Central)	ca-central-1
Europe (Frankfurt)	eu-central-1
Europe (Stockholm)	eu-north-1
Europe (Ireland)	eu-west-1
Europe (London)	eu-west-2
Europe (Paris)	eu-west-3
South America (São Paulo)	sa-east-1
US East (N. Virginia)	us-east-1
US East (Ohio)	us-east-2
US West (N. California)	us-west-1
US West (Oregon)	us-west-2

For a list of Region names and their corresponding codes, see Regional endpoints in the *AWS General Reference Guide*. For a list of AWS services supported in each Region (without endpoints), see the AWS Regional Services List.

Important

AWS recommends that you use regional AWS Security Token Service (AWS STS) endpoints instead of the global endpoint to reduce latency. Session tokens from regional AWS STS endpoints are valid in all AWS Regions. If you use regional AWS STS endpoints, you don't need to make any changes. However, session tokens from the *global* AWS STS endpoint (https://sts.amazonaws.com) are valid only in AWS Regions that you enable, or that are enabled by default. If you intend to enable a new Region for your account, you can either use session tokens from regional AWS STS endpoints or activate the global AWS STS endpoint to issue session tokens that are valid in all AWS Regions. Session tokens that are valid in all Regions are larger. If you store session tokens, these larger tokens might affect your systems. For more information about how AWS STS endpoints work with AWS Regions, see Managing AWS STS in an AWS Region.

Considerations before enabling and disabling Regions

Before you enable or disable a Region, it's important to consider the following:

- **You can use all destination Regions in a cross-Region inference geography regardless of Region-opt status** – Certain AWS generative AI services including Amazon Bedrock (see Increase throughput with cross-Region inference) and Amazon Q Developer (see Cross-region processing in Amazon Q Developer) use cross-region inference. If you use those services, they automatically select the optimal AWS Region–including Regions that you have not enabled for resources and IAM data–within your chosen geography. This improves the customer experience by maximizing available compute and model availability.
- **You can use IAM permissions to control access to Regions** – AWS Identity and Access Management (IAM) includes four permissions that let you control which users can enable, disable, get, and list Regions. For more information, see AWS: Allows enabling and disabling AWS Regions in the *IAM User Guide*. You can also use the `aws:RequestedRegion` condition key to control access to AWS services in an AWS Region.
- **Enabling and disabling a Region is free** – There is no charge to enable or disable a Region. You are charged only for resources that you create in the new Region.
- **Amazon EventBridge integration** – You can subscribe to region-opt status update notifications in EventBridge. An EventBridge notification will be created for each status change, allowing customers to automate work flows.

- **Expressive Region-opt status** – Due to the asynchronous nature of enabling/disabling an opt-in region, there are four potential statuses for a region-opt request:
  - `ENABLING`
  - `DISABLING`
  - `ENABLED`
  - `DISABLED`

  You cannot cancel an opt-in or opt-out when it is in either `ENABLING` or `DISABLING` status. Otherwise, a `ConflictException` will be thrown. A completed (Enabled/Disabled) region-opt request is dependent on the provisioning of key underlying AWS services. There might be some AWS services that will not be immediately usable despite the status being `ENABLED`.

Processing times and request limits

When enabling or disabling Regions, be aware of the following timing and request limitations:

- **Enabling a Region takes a few minutes to several hours in some cases** – When you enable a Region, AWS performs actions to prepare your account in that Region, such as distributing your IAM resources to the Region. This process takes a few minutes for most accounts, but can sometimes take several hours. You cannot use the Region until this process is complete.
- **Disabling a Region isn't always immediately visible** – Services and consoles might be temporarily visible after disabling a region. Disabling a Region can take a few minutes to several hours to take effect.
- **A single account can have 6 region-opt requests in progress at any given time** – One request is equal to either an enable or disable of one particular region for one account.
- **Organizations can have 50 region-opt requests open at a given time across an AWS organization** – The management account can at any point in time have 50 open requests pending completion for its organization. One request is equal to either an enable or disable of one particular region for one account.

Enable or disable a Region for standalone accounts

To update which Regions your AWS account has access to, perform the steps in the following procedure. The following AWS Management Console procedure always works only in the standalone context. You can use the AWS Management Console to view or update only the available Regions in the account you used to call the operation.

AWS Management Console

To enable or disable a Region for a standalone AWS account

> **Minimum permissions**
>
> To perform the steps in the following procedure, an IAM user or role must have the following permissions:
>
> - `account:ListRegions` (needed to view the list of AWS Regions and whether they are currently enabled or disabled).
> - `account:EnableRegion`
> - `account:DisableRegion`

1. Sign in to the AWS Management Console as either the AWS account root user or as an IAM user or role that has the minimum permissions.
2. Choose your account name on the top right of the window, and then choose **Account**.
3. On the **Account** page, scroll down to the section **AWS Regions**.
4. Choose the Region that you want to enable or disable and then choose the desired action **Enable** or **Disable**. You will see a prompt to confirm.
5. If you chose the **Enable** option, review the displayed text and then choose **Enable region**.

   If you chose the **Disable** option, review the displayed text, type **`disable`** to confirm, and then choose **Disable region**.

   After the opt-in Region is enabled, you can select that Region from the Region navigation bar. For steps to select a Region, see Choosing a Region from the navigation bar in the AWS Management Console and for Region specific console setting in your account, see Setting the default Region in the AWS Management Console.

AWS CLI & SDKs

You can enable, disable, read and list region opt status by using the following AWS CLI commands or their AWS SDK equivalent operations:

- `EnableRegion`

- `DisableRegion`
- `GetRegionOptStatus`
- `ListRegions`

> **Minimum permissions**
>
> To perform the following steps, you must have the permission that maps to that operation:
>
> - `account:EnableRegion`
> - `account:DisableRegion`
> - `account:GetRegionOptStatus`
> - `account:ListRegions`

If you use these individual permissions, you can grant some users the ability to only read region opt information, and grant others the ability to both read and write.

The following example enables a region for the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

Note that you can also disable a region using the same command and then replacing `enable-region` with `disable-region`.

```
aws account enable-region --region-name af-south-1
```

This command produces no output if it's successful.

The operation is asynchronous. The following command will allow you to see the latest status of the request.

```
aws account get-region-opt-status --region-name af-south-1
  {
    "RegionName": "af-south-1",
    "RegionOptStatus": "ENABLING"
  }
```

Enable or disable a Region in your organization

To update the enabled Regions for member accounts of your AWS Organizations, perform the steps in the following procedure.

> **Note**
>
> The AWS Organizations managed policies `AWSOrganizationsReadOnlyAccess` or `AWSOrganizationsFullAccess` are updated to provide permission to access the AWS Account Management APIs so you can access account data from the AWS Organizations console. To view the updated managed policies, see Updates to Organizations AWS managed policies.

> **Note**
>
> Before you can perform these operations from the management account or a delegated admin account in an organization for use with member accounts, you must:
>
> - Enable all features in your organization to manage settings on your member accounts. This allows admin control over the member accounts. This is set by default when you create your organization. If your organization is set to consolidated billing only, and you want to enable all features, see Enabling all features in your organization.
> - Enable trusted access for the AWS Account Management service. To set this up, see Enable trusted access for AWS Account Management.

AWS Management Console

To enable or disable a Region in your organization

1. Sign in to the AWS Organizations console with your organization's management account credentials.
2. On the **AWS accounts** page, select the account that you want to update.
3. Choose the **Account settings** tab.
4. Under **Regions**, select the Region you want to enable or disable.
5. Choose **Actions**, and then choose either **Enable** or **Disable** option.

6. If you chose the **Enable** option, review the displayed text and then choose **Enable region**.
7. If you chose the **Disable** option, review the displayed text, type **disable** to confirm, and then choose **Disable region**.

AWS CLI & SDKs

You can enable, disable, read and list region opt status for organization member accounts by using the following AWS CLI commands or their AWS SDK equivalent operations:

- `EnableRegion`
- `DisableRegion`
- `GetRegionOptStatus`
- `ListRegions`

> **Minimum permissions**
>
> To perform the following steps, you must have the permission that maps to that operation:
>
> - `account:EnableRegion`
> - `account:DisableRegion`
> - `account:GetRegionOptStatus`
> - `account:ListRegions`

If you use these individual permissions, you can grant some users the ability to only read region opt information, and grant others the ability to both read and write.

The following example enables a region for the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

Note that you can also disable a region using the same command and then replacing `enable-region` with `disable-region`.

```
aws account enable-region --account-id 123456789012 --region-name af-south-1
```

This command produces no output if it's successful.

> **Note**
>
> An organization can only have up to 20 region requests at a given time. Otherwise, you will receive a `TooManyRequestsException`.

The operation is asynchronous. The following command will allow you to see the latest status of the request.

```
aws account get-region-opt-status --account-id 123456789012 --region-name af-south-1
  {
    "RegionName": "af-south-1",
    "RegionOptStatus": "ENABLING"
  }
```

Update billing for your AWS account

These instructions are for how to update billing for your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

You can update all of your AWS account billing preferences using the AWS Billing and Cost Management console. To learn how to update billing-related settings for your account, see the *AWS Billing and Cost Management User Guide*:

Update the root user email address

These instructions are for how to update the root user email address if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

There are various business reasons why you might need to update the root user email address of your AWS account. For example, security and administrative resilience. This topic walks you through the process of updating your root user email address for both standalone and member accounts.

> **Note**
>
> Changes to an AWS account can take up to four hours to propagate everywhere.

You can update the root user email differently, depending on whether or not the accounts are standalone, or part of an organization:

- **Standalone AWS accounts or management accounts** – For AWS accounts not associated with an organization, or for the management account of an organization, you can update the root user email using the AWS Management Console. To learn how to do this, see Update the root user email for a standalone AWS account or management account.
- **AWS accounts within an organization** – For member accounts that are part of an AWS organization, a user in the management account or delegated admin account can centrally update the root user email of the member account from the AWS Organizations console, or programmatically via the AWS CLI & SDKs. To learn how to do this, see Update the root user email for any AWS account in your organization.

Topics

Update the root user email for a standalone AWS account or management account

To edit the root user email address for a standalone AWS account or a management account, perform the steps in the following procedure.

AWS Management Console

> **Note**
>
> You must sign in as the AWS account root user, which requires no additional IAM permissions. You can't perform these steps as an IAM user or role.

1. Use your AWS account's email address and password to sign in to the AWS Management Console as your AWS account root user.
2. In the upper right corner of the console, choose your account name or number and then choose **Account**.
3. On the **Account** page, next to **Account details**, choose **Actions** and then select **Update email address and password**.
4. On the **Account Details** page, next to **Email address** choose **Edit**.
5. On the **Edit Account Email** page, fill out the fields for **New email address**, **Confirm new email address**, and confirm your current **Password**. Then, choose **Save and continue**. A verification code is sent to your new email address from `no-reply@verify.signin.aws`.
6. On the **Edit Account Email** page, under **Verification code**, enter the code you received from your email, and then choose **Confirm updates**.

> **Note**
>
> It can take up to 5 minutes for the verification code to arrive. If you don't see the email in your inbox, check your spam and junk folders.

AWS CLI & SDKs

This task isn't supported in the AWS CLI or by an API operation from one of the AWS SDKs. You can perform this task only by using the AWS Management Console.

Update the root user email for any AWS account in your organization

To edit the root user email address for any member account in your organization using the AWS Organizations console, perform the steps in the following procedure.

> **Note**
>
> Before you update the root user email address for a member account, we recommend that you understand the impact of this operation. For more information, see Updating the root user email address for a member account with AWS Organizations in the *AWS Organizations User Guide*.

You can also update the root user email address for a member account directly from the **Account** page in the AWS Management Console after signing in as the root user. For step-by-step instructions, follow the steps provided in Update the root user email for a standalone AWS account or management account.

AWS Management Console

Notes

- To perform this procedure from the management account or a delegated admin account in an organization against member accounts, you must enable trusted access for the Account Management service.
- You can't use this procedure to access an account in a different organization from the one you are using to call the operation.

To update the root user email address for a member account using the AWS Organizations console

1. Sign in to the AWS Organizations console. You must sign in as an IAM user, or sign in as the root user (not recommended) in the organization's management account.
2. On the **AWS accounts** page, choose the member account for which you want to update the root user email address.
3. In the **Account details** section, choose the **Actions** button, and then choose **Update email address**.
4. Under **Email**, enter the new email address for the root user, and then choose **Save**. This sends a one-time password (OTP) to the new email address.

Note

If you need to close this page in the Organizations console while you wait for the code, you can return and finish the OTP process within 24 hours from when the code was sent. To do this, while on the **Account details** page, choose the **Actions** button, and then choose **Complete email update**.

5. Under **Verification code**, enter the code that was sent to the new email address in the previous step, and then choose **Confirm**. This commits the update to the root user for the account.

AWS CLI & SDKs

You can retrieve, or update the ***root user*** email address (also referred to as the primary email address) by using the following AWS CLI commands or their AWS SDK equivalent operations:

- GetPrimaryEmail
- StartPrimaryEmailUpdate
- AcceptPrimaryEmailUpdate

Notes

- To perform these operations from the management account or a delegated admin account in an organization against member accounts, you must enable trusted access for the Account Management service.
- You can't access an account in a different organization from the one you are using to call the operation.

Minimum permissions

For each operation, you must have the permission that maps to that operation:

- `account:GetPrimaryEmail`
- `account:StartPrimaryEmailUpdate`
- `account:AcceptPrimaryEmailUpdate`

If you use these individual permissions, you can grant some users the ability to only read the root user email address information, and grant others the ability to both read and write.

To complete the root user email address process, you must use the primary email APIs together in the order they are shown in the following examples.

Example `GetPrimaryEmail`

The following example retrieves the root user email address from the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account get-primary-email --account-id 123456789012
```

Example `StartPrimaryEmailUpdate`

The following example starts the root user email address update process, identifies the new email address, and sends a one-time password (OTP) to the new email address for the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account start-primary-email-update --account-id 123456789012 --primary-email
 john@examplecorp.com
```

Example `AcceptPrimaryEmailUpdate`

The following example accepts the OTP code and sets the new email address to the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account accept-primary-email-update --account-id 123456789012 --otp 12345678
 --primary-email john@examplecorp.com
```

Update root user password

These instructions are for how to update the root user password if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

To edit your AWS account's root user password, perform the steps in the following procedure.

AWS Management Console

To edit your root user password

> **Note**
>
> You must sign in as the AWS account root user, which requires no additional IAM permissions. You can't perform these steps as an IAM user or role.

1. Use your AWS account's email address and password to sign in to the AWS Management Console as your AWS account root user.
2. In the upper right corner of the console, choose your account name or number and then choose **Account**.
3. On the **Account** page, next to **Account details**, choose **Actions** and then select **Update email address and password**.
4. On the **Account Details** page, next to **Password** choose **Edit**.
5. On the **Edit password** page, fill out the fields for **Current password**, **New password**, and **Confirm new password**. Then, choose **Update password**. For additional guidance including best practices for setting root user passwords, see Change the password for the AWS account root user in the *IAM User Guide*.

AWS CLI & SDKs

This task isn't supported in the AWS CLI or by an API operation from one of the AWS SDKs. You can perform this task only by using the AWS Management Console.

Update your AWS account name

These instructions are for how to update your AWS account name if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

When managing multiple AWS accounts, use clear naming conventions aligned with business units and applications for identification and organization. During reorganizations, mergers, acquisitions,

or naming convention updates, you might need to rename accounts to maintain consistent identification and administrative standards.

The name of an account appears in several places, such as on your invoice and in consoles such as the Billing and Cost Management dashboard and the AWS Organizations console. We recommend that you use a standard way to name your accounts so that your account names are easy to recognize. For company accounts, consider using a naming standard such as *organization-purpose-environment* (for example, *sales-catalog-prod*). For privacy and security reasons, avoid using account names that reflect personally identifiable information (PII).

- **Standalone AWS accounts or management accounts –** For AWS accounts not associated with an organization, or for the management account of an organization, you can update your account name using the AWS Management Console, or the AWS CLI and SDKs. To learn how to do this, see Update your account name for a standalone AWS account or management account.
- **AWS accounts within an organization –** For member accounts that are part of an AWS Organizations, a user in the management account or delegated admin account can centrally update the account name of any member account in the organization from the AWS Organizations console, or programmatically via the AWS CLI and SDKs. To learn how to do this, see Update your account name for any AWS account in your organization.

> **Note**
>
> Changes to an AWS account can take up to four hours to propagate everywhere.

Topics

Update your account name for a standalone AWS account or management account

To change the account name for a standalone AWS account or a management account, perform the steps in the following procedure.

AWS Management Console

Minimum permissions

You can update your account name using the root user, an IAM user, or an IAM role. If you are using the root user, no additional IAM permissions are needed to update an account name. When using an IAM user or IAM role, you must have at least the following IAM permissions:

- `account:GetAccountInformation`
- `account:PutAccountName`

To update the account name for a standalone account or management account

1. Use your AWS account's email address and password to sign in to the AWS Management Console as your AWS account root user.
2. In the upper right corner of the console, choose your account name or number and then choose **Account**.
3. On the **Account** page, next to **Account details**, choose **Actions** and then select **Update account name**.
4. Under **Name**, enter the new account name you want to update, and then choose **Save**.

AWS CLI & SDKs

Minimum permissions

You can update your account name using the root user, an IAM user, or an IAM role. To perform the following steps, your IAM user or IAM role must have at least the following IAM permissions:

- `account:GetAccountInformation`
- `account:PutAccountName`

To update the account name for a standalone account or management account

You can use one of the following operations:

- AWS CLI: put-account-name

```
$ C:\> aws account put-account-name \
        --account-name "New-Account-Name"
```

- AWS SDKs: PutAccountName

Update your account name for any AWS account in your organization

In AWS Organizations with all features mode, authorized IAM users or IAM roles in both management and delegated admin accounts can centrally manage account names.

To change the account name for any member account in your organization, perform the steps in the following procedure.

Requirements

To update an account name with the AWS Organizations console, you need to do some preliminary settings:

- Your organization must enable *all features* to manage settings on your member accounts. This allows admin control over the member accounts. This is set by default when you create your organization. If your organization is set to *consolidated billing* only, and you want to enable all features, see Enabling all features for an organization.
- You need to enable trusted access for the AWS Account Management service. To set this up, see Enable trusted access for AWS Account Management.

AWS Management Console

Minimum permissions

To update the account name for a member account, your IAM user or IAM role must have the following permissions:

- `organizations:DescribeOrganization` (console only)
- `account:PutAccountName`

To update the account name for a member account

1. Open the Organizations console at https://console.aws.amazon.com/organizations/.
2. In the main navigation pane, choose **AWS accounts**.
3. On the **AWS accounts** page, choose the member account that you want to update, choose the **Actions** drop-down menu, and then choose **Update account name**.
4. Under **Name**, enter the updated name, and choose **Save**.

AWS CLI & SDKs

> **Minimum permissions**
>
> To update the account name for a member account, your IAM user or IAM role must have the following permissions:
>
> - `organizations:DescribeOrganization` (console only)
> - `account:PutAccountName`

To update the account name for a member account

You can use one of the following operations:

- AWS CLI: put-account-name

```
$ C:\> aws account put-account-name \
        --account-id 111111111111 \
        --account-name "New-Account-Name"
```

- AWS SDKs: PutAccountName

Update the alternate contacts for your AWS account

These instructions are for how to update the alternate contacts for your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

Alternate contacts allows AWS to contact up to three alternate contacts associated with the account. An alternate contact doesn't have to be a specific person. You could instead add an email distribution list if you have a team that manages billing, operations and security related issues. These are in addition to the email address associated with the root user of the account. The primary account contact will continue to receive all email communications sent to the root account's email.

You can specify only one of each of the following contact types associated with an account.

- Billing contact
- Operations contact
- Security contact

You can add or edit alternate contacts differently, depending on whether or not the accounts are standalone, or part of an organization:

- **Standalone AWS accounts** – For AWS accounts not associated with an organization, you can update your own alternate contacts using the AWS Management Console, or via AWS CLI & SDKs. To learn how to do this, see Update the alternate contacts for a standalone AWS account.
- **AWS accounts within an organization** – For member accounts that are part of an AWS organization, a user in the management account or delegated admin account can centrally update any member account in the organization from the AWS Organizations console, or programmatically via the AWS CLI & SDKs. To learn how to do this, see Update the alternate contacts for any AWS account in your organization.

Topics

Phone number and email address requirements

Before you proceed with updating your account's alternate contacts information, we recommend that you first review the following requirements when entering phone numbers and email addresses.

- Phone numbers can only contain numbers, whitespaces and the following characters:"+ - ( )".
- Email addresses can be up to 254 characters long and can include the following special characters in the local portion of the email address in addition to the standard alphanumeric ones: "+ = . # | ! & - _".

Update the alternate contacts for a standalone AWS account

To add or edit the alternate contacts for a standalone AWS account, perform the steps in the following procedure. The following AWS Management Console procedure always works *only* in the standalone context. You can use the AWS Management Console to access or change only the alternate contacts in the account you used to call the operation.

AWS Management Console

To add or edit the alternate contacts for a standalone AWS account

> **Minimum permissions**
>
> To perform the following steps, you must have at least the following IAM permissions:
>
> - `account:GetAlternateContact` (to see the alternate contact details)
> - `account:PutAlternateContact` (to set or update an alternate contact)
> - `account:DeleteAlternateContact` (to delete an alternate contact)

1. Sign in to the AWS Management Console as an IAM user or role that has the minimum permissions.
2. Choose your account name on the top right of the window, and then choose **Account**.
3. On the **Account** page, scroll down to **Alternate contacts**, and to the right of the title, choose **Edit**.

> **Note**
>
> If you don't see the **Edit** option, it is likely that you are not signed in as the root user for your account or as someone who has the minimum permissions specified previously.

4. Change the values in any of the available fields.

> **Important**
>
> For business AWS accounts, it's a best practice to enter a company phone number and email address rather than one belonging to an individual.

5. After you have made all of your changes, choose **Update**.

AWS CLI & SDKs

You can retrieve, update, or delete the ***alternate*** contact information by using the following AWS CLI commands or their AWS SDK equivalent operations:

- GetAlternateContact
- PutAlternateContact
- DeleteAlternateContact

> **Notes**
>
> - To perform these operations from the management account or a delegated admin account in an organization against member accounts, you must enable trusted access for the Account service.

> **Minimum permissions**
>
> For each operation, you must have the permission that maps to that operation:
>
> - `GetAlternateContact` (to see the alternate contact details)

- `PutAlternateContact` (to set or update an alternate contact)
- `DeleteAlternateContact` (to delete an alternate contact)

If you use these individual permissions, you can grant some users the ability to only read the contact information, and grant others the ability to both read and write.

Example

The following example retrieves the current Billing alternate contact for the caller's account.

```
$ aws account get-alternate-contact \
    --alternate-contact-type=BILLING
{
    "AlternateContact": {
        "AlternateContactType": "BILLING",
        "EmailAddress": "saanvi.sarkar@amazon.com",
        "Name": "Saanvi Sarkar",
        "PhoneNumber": "+1(206)555-0123",
        "Title": "CFO"
    }
}
```

Example

The following example sets a new Operations alternate contact for the caller's account.

```
$ aws account put-alternate-contact \
    --alternate-contact-type=OPERATIONS \
    --email-address=mateo_jackson@amazon.com \
    --name="Mateo Jackson" \
    --phone-number="+1(206)555-1234" \
    --title="Operations Manager"
```

This command produces no output if it's successful.

Example

> **Note**
>
> If you perform multiple `PutAlternateContact` operations on the same AWS account and the same contact type, the first adds the new contact, and all successive calls to the same AWS account and contact type update the existing contact.

Example

The following example deletes the Security alternate contact for the caller's account.

```
$ aws account delete-alternate-contact \
    --alternate-contact-type=SECURITY
```

This command produces no output if it's successful.

> **Note**
>
> If you try to delete the same contact more than once, the first succeeds silently. All later attempts generate a `ResourceNotFound` exception.

Update the alternate contacts for any AWS account in your organization

To add or edit the alternate contact details for any AWS account in your organization, perform the steps in the following procedure.

Requirements

To update alternate contacts with the AWS Organizations console, you need to do some preliminary settings:

- Your organization must enable *all features* to manage settings on your member accounts. This allows admin control over the member accounts. This is set by default when you create your organization. If your organization is set to *consolidated billing* only, and you want to enable all features, see Enabling all features for an organization.
- You need to enable trusted access for the AWS Account Management service. To set this up, see Enable trusted access for AWS Account Management.

> **Note**
>
> The AWS Organizations managed policies `AWSOrganizationsReadOnlyAccess` or `AWSOrganizationsFullAccess` are updated to provide permission to access the AWS Account Management APIs so you can access account data from the AWS Organizations console. To view the updated managed policies, see Updates to Organizations AWS managed policies.

AWS Management Console

To add or edit the alternate contacts for any AWS account in your organization

1. Sign in to the AWS Organizations console with the organization's management account credentials.
2. From **AWS accounts**, select the account that you want to update.
3. Choose **Contact info**, and under **Alternate contacts**, locate the type of contact: **Billing contact**, **Security contact**, or **Operations contact**.
4. To add a new contact, select **Add**, or to update an existing contact select **Edit**.
5. Change the values in any of the available fields.

   > **Important**
   >
   > For business AWS accounts, it's a best practice to enter a company phone number and email address rather than one belonging to an individual.

6. After you have made all of your changes, choose **Update**.

AWS CLI & SDKs

You can retrieve, update, or delete the ***alternate*** contact information by using the following AWS CLI commands or their AWS SDK equivalent operations:

- GetAlternateContact
- PutAlternateContact
- DeleteAlternateContact

Notes

- To perform these operations from the management account or a delegated admin account in an organization against member accounts, you must enable trusted access for the Account service.
- You can't access an account in a different organization from the one you are using to call the operation.

Minimum permissions

For each operation, you must have the permission that maps to that operation:

- `GetAlternateContact` (to see the alternate contact details)
- `PutAlternateContact` (to set or update an alternate contact)
- `DeleteAlternateContact` (to delete an alternate contact)

If you use these individual permissions, you can grant some users the ability to only read the contact information, and grant others the ability to both read and write.

Example

The following example retrieves the current Billing alternate contact for the caller's account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account get-alternate-contact \
    --alternate-contact-type=BILLING \
    --account-id 123456789012
{
    "AlternateContact": {
        "AlternateContactType": "BILLING",
        "EmailAddress": "saanvi.sarkar@amazon.com",
        "Name": "Saanvi Sarkar",
```

```
        "PhoneNumber": "+1(206)555-0123",
        "Title": "CFO"
    }
}
```

Example

The following example sets the Operations alternate contact for the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account put-alternate-contact \
    --account-id 123456789012 \
    --alternate-contact-type=OPERATIONS \
    --email-address=mateo_jackson@amazon.com \
    --name="Mateo Jackson" \
    --phone-number="+1(206)555-1234" \
    --title="Operations Manager"
```

This command produces no output if it's successful.

> **Note**
>
> If you perform multiple `PutAlternateContact` operations on the same AWS account and the same contact type, the first adds the new contact, and all successive calls to the same AWS account and contact type update the existing contact.

Example

The following example deletes the Security alternate contact for the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account delete-alternate-contact \
    --account-id 123456789012 \
    --alternate-contact-type=SECURITY
```

This command produces no output if it's successful.

Example

> **Note**
>
> If you try to delete the same contact more than once, the first succeeds silently. All later attempts generate a `ResourceNotFound` exception.

account:AlternateContactTypes context key

You can use the context key `account:AlternateContactTypes` to specify which of the three contact types is allowed (or denied) by the IAM policy. For example, the following example IAM permission policy uses this condition key to allow the attached principals to retrieve, but not modify, only the `BILLING` alternate contact for a specific account in an organization.

Because `account:AlternateContactTypes` is a multi-valued string type, you must use the `ForAnyValue` or `ForAllValues` multi-value string operators.

Update the primary contact for your AWS account

These instructions are for how to update the primary contact for your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

You can update the primary contact information associated with your account, including your contact's full name, company name, mailing address, telephone number, and website address.

You edit the primary account contact differently, depending on whether or not the accounts are standalone, or part of an organization:

- **Standalone AWS accounts** – For AWS accounts not associated with an organization, you can update your own primary account contact using the AWS Management Console, or via AWS CLI & SDKs. To learn how to do this, see Update standalone AWS account primary contact.
- **AWS accounts within an organization** – For member accounts that are part of an AWS organization, a user in the management account or delegated admin account can centrally update any member account in the organization from the AWS Organizations console, or programmatically via the AWS CLI & SDKs. To learn how to do this, see Update AWS account primary contact in your organization.

Topics

Phone number and email address requirements

Before you proceed with updating your account's primary contact information, we recommend that you first review the following requirements when entering phone numbers and email addresses.

- Phone numbers should only contain numbers.
- Phone numbers must start with a + and country code and must not have any leading zeros or additional spaces after the country code. For example, +1 (US/Canada) or +44 (UK).
- Phone numbers must not include whitespaces between the area code, exchange code, and local code. For example, +12025550179.
- For security purposes, phone numbers must be capable of receiving SMS from AWS. Toll free numbers will not be accepted since most don't support SMS.
- For business AWS accounts, it's a best practice to enter a company phone number and email address rather than one belonging to an individual. Configuring the account's root user with an individual's email address or phone number can make your account difficult to recover if that individual leaves the company.

Update the primary contact for a standalone AWS account or management account

To edit your primary contact details for a standalone AWS account or a management account, perform the steps in the following procedure. The following AWS Management Console procedure always works *only* in the standalone context. You can use the AWS Management Console to access or change only the primary contact information of the account you used to call the operation.

AWS Management Console

To edit your primary contact for a standalone AWS account or management account

> **Minimum permissions**
>
> To perform the following steps, you must have at least the following IAM permissions:
>
> - `account:GetContactInformation` (to see the primary contact details)
> - `account:PutContactInformation` (to update the primary contact details)

1. Sign in to the AWS Management Console as an IAM user or role that has the minimum permissions.
2. Choose your account name on the top right of the window, and then choose **Account**.
3. Scroll down to the section **Contact information**, and next to it choose **Edit**.
4. Change the values in any of the available fields.
5. After you have made all of your changes, choose **Update**.

AWS CLI & SDKs

You can retrieve, update, or delete the ***primary*** contact information by using the following AWS CLI commands or their AWS SDK equivalent operations:

- GetContactInformation
- PutContactInformation

> **Notes**
>
> - To perform these operations from the management account or a delegated admin account in an organization against member accounts, you must enable trusted access for the Account service.

Minimum permissions

For each operation, you must have the permission that maps to that operation:

- `account:GetContactInformation`
- `account:PutContactInformation`

If you use these individual permissions, you can grant some users the ability to only read the contact information, and grant others the ability to both read and write.

Example

The following example retrieves the current primary contact information for the caller's account.

```
$ aws account get-contact-information
{
    "ContactInformation": {
        "AddressLine1": "123 Any Street",
        "City": "Seattle",
        "CompanyName": "Example Corp, Inc.",
        "CountryCode": "US",
        "DistrictOrCounty": "King",
        "FullName": "Saanvi Sarkar",
        "PhoneNumber": "+15555550100",
        "PostalCode": "98101",
        "StateOrRegion": "WA",
        "WebsiteUrl": "https://www.examplecorp.com"
    }
}
```

Example

The following example sets new primary contact information for the caller's account.

```
$ aws account put-contact-information --contact-information \
'{"AddressLine1": "123 Any Street", "City": "Seattle", "CompanyName": "Example Corp,
 Inc.", "CountryCode": "US", "DistrictOrCounty": "King",
```

```
"FullName": "Saanvi Sarkar", "PhoneNumber": "+15555550100", "PostalCode": "98101",
 "StateOrRegion": "WA", "WebsiteUrl": "https://www.examplecorp.com"}'
```

This command produces no output if it's successful.

Update the primary contact for any AWS member account in your organization

To edit your primary contact details in any AWS member account in your organization, perform the steps in the following procedure.

Additional requirements

To update primary contact with the AWS Organizations console, you need to do some preliminary settings:

- Your organization must enable *all features* to manage settings on your member accounts. This allows admin control over the member accounts. This is set by default when you create your organization. If your organization is set to *consolidated billing* only, and you want to enable all features, see Enabling all features for an organization.
- You need to enable trusted access for the AWS Account Management service. To set this up, see Enable trusted access for AWS Account Management.

AWS Management Console

To edit your primary contact for any AWS account in your organization

1. Sign in to the AWS Organizations console with the organization's management account credentials.
2. From **AWS accounts**, select the account that you want to update.
3. Choose **Contact info**, and locate **Primary contact**,
4. Select **Edit**.
5. Change the values in any of the available fields.
6. After you have made all of your changes, choose **Update**.

AWS CLI & SDKs

You can retrieve, update, or delete the ***primary*** contact information by using the following AWS CLI commands or their AWS SDK equivalent operations:

- GetContactInformation
- PutContactInformation

Notes

- To perform these operations from the management account or a delegated admin account in an organization against member accounts, you must enable trusted access for the Account service.
- You can't access an account in a different organization from the one you are using to call the operation.

Minimum permissions

For each operation, you must have the permission that maps to that operation:

- `account:GetContactInformation`
- `account:PutContactInformation`

If you use these individual permissions, you can grant some users the ability to only read the contact information, and grant others the ability to both read and write.

Example

The following example retrieves the current primary contact information for the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account get-contact-information --account-id 123456789012
{
```

```
    "ContactInformation": {
        "AddressLine1": "123 Any Street",
        "City": "Seattle",
        "CompanyName": "Example Corp, Inc.",
        "CountryCode": "US",
        "DistrictOrCounty": "King",
        "FullName": "Saanvi Sarkar",
        "PhoneNumber": "+15555550100",
        "PostalCode": "98101",
        "StateOrRegion": "WA",
        "WebsiteUrl": "https://www.examplecorp.com"
    }
}
```

Example

The following example sets the primary contact information for the specified member account in an organization. The credentials used must be from either the organization's management account, or from the Account Management's delegated admin account.

```
$ aws account put-contact-information --account-id 123456789012 \
--contact-information '{"AddressLine1": "123 Any Street", "City": "Seattle",
 "CompanyName": "Example Corp, Inc.", "CountryCode": "US", "DistrictOrCounty":
 "King",
"FullName": "Saanvi Sarkar", "PhoneNumber": "+15555550100", "PostalCode": "98101",
 "StateOrRegion": "WA", "WebsiteUrl": "https://www.examplecorp.com"}'
```

This command produces no output if it's successful.

View AWS account identifiers

This information is about viewing AWS account identifiers if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

AWS assigns the following unique identifiers to each AWS account:

AWS account ID

A 12-digit number, such as 012345678901, that uniquely identifies an AWS account. Many AWS resources include the account ID in their Amazon Resource Names (ARNs). The account ID

portion distinguishes resources in one account from the resources in another account. If you are an AWS Identity and Access Management (IAM) user, you can sign in to the AWS Management Console using either the account ID or account alias. While account IDs, like any identifying information, should be used and shared carefully, they are not considered secret, sensitive, or confidential information.

Canonical user ID

An alpha-numeric identifier, such as `79a59df900b949e55d96a1e698fbacedfd6e09d98eacf8f8d5218e7cd47ef2be`, that is an obfuscated form of the AWS account ID. You can use this ID to identify an AWS account when granting cross-account access to buckets and objects using Amazon Simple Storage Service (Amazon S3). You can retrieve the canonical user ID for your AWS account as either the root user or an IAM user.

You must be authenticated with AWS to view these identifiers.

> **Warning**
>
> Do not provide your AWS credentials (including passwords and access keys) to a third party that needs your AWS account identifiers to share AWS resources with you. Doing so would give them the same access to the AWS account that you have.

Find your AWS account ID

You can find the AWS account ID using either the AWS Management Console or the AWS Command Line Interface (AWS CLI). In the console, the location of the account ID depends on whether you are signed in as the root user or an IAM user. The account ID is the same whether you are signed in as the root user or an IAM user.

Finding your account ID as the root user

AWS Management Console

To find your AWS account ID when signed in as the root user

> **Minimum permissions**
>
> To perform the following steps, you must have at least the following IAM permissions:

- When you sign in as the root user, you don't need any IAM permissions.

1. In the navigation bar on the upper right, choose your account name or number and then choose **Security credentials**.

   > **Tip**
   >
   > If you don't see the **Security credentials** option, you might be signed in as a federated user with an IAM role, instead of as an IAM user. In this case, look for the entry **Account** and the account ID number next to it.

2. Under the **Account details** section, the account number appears next to **AWS account ID**.

AWS CLI & SDKs

To find your AWS account ID using the AWS CLI

> **Minimum permissions**
>
> To perform the following steps, you must have at least the following IAM permissions:
>
> - When you run the command as the root user, you don't need any IAM permissions.

Use the get-caller-identity command as follows.

```
$ aws sts get-caller-identity \
    --query Account \
    --output text
123456789012
```

Find your account ID as an IAM user

AWS Management Console

To find your AWS account ID when signed in as an IAM user

> **Minimum permissions**
>
> To perform the following steps, you must have at least the following IAM permissions:
>
> - `account:GetAccountInformation`

1. In the navigation bar on the upper right, choose your user name and then choose **Security credentials**.

   > **Tip**
   >
   > If you don't see the **Security credentials** option, you might be signed in as a federated user with an IAM role, instead of as an IAM user. In this case, look for the entry **Account** and the account ID number next to it.

2. At the top of the page, under **Account details**, the account number appears next to **AWS account ID**.

AWS CLI & SDKs

To find your AWS account ID using the AWS CLI

> **Minimum permissions**
>
> To perform the following steps, you must have at least the following IAM permissions:
>
> - When you run the command as an IAM user or role, then you must have:
>   - `sts:GetCallerIdentity`

Use the get-caller-identity command as follows.

```
$ aws sts get-caller-identity \
```

```
    --query Account \
    --output text
123456789012
```

Find the canonical user ID for your AWS account

You can find the canonical user ID for your AWS account using the AWS Management Console or the AWS CLI. The canonical user ID for an AWS account is specific to that account. You can retrieve the canonical user ID for your AWS account as the root user, a federated user, or an IAM user.

Find the canonical ID as the root user or IAM user

AWS Management Console

To find the canonical user ID for your account when signed in to the console as the root user or an IAM user

Minimum permissions

To perform the following steps, you must have at least the following IAM permissions:

- When you run the command as the root user, you don't need any IAM permissions.
- When you sign in as an IAM user, then you must have:
  - `account:GetAccountInformation`

1. Sign in to the AWS Management Console as the root user or an IAM user.
2. In the navigation bar on the upper right, choose your account name or number and then choose **Security credentials**.

   Tip

   If you don't see the **Security credentials** option, you might be signed in as a federated user with an IAM role, instead of as an IAM user. In this case, look for the entry **Account** and the account ID number next to it.
3. Under the **Account details** section, the canonical user ID appears next to **Canonical user ID**. You can use your canonical user ID to configure Amazon S3 access control lists (ACLs).

AWS CLI & SDKs

To find the canonical user ID using the AWS CLI

The same AWS CLI and API command works for the AWS account root user, IAM users, or IAM roles.

Use the list-buckets command as follows.

```
$ aws s3api list-buckets \
    --max-items 10 \
    --page-size 10 \
    --query Owner.ID \
    --output text
249fa2f1dc32c330EXAMPLE91b2778fcc65f980f9172f9cb9a5f50ccbEXAMPLE
```

Find the canonical ID as a federated user with an IAM role

AWS Management Console

To find the canonical ID for your account when signed in to the console as a federated user with an IAM role

> **Minimum permissions**
>
> - You must have permission to list and view an Amazon S3 bucket.

1. Sign in to the AWS Management Console as a federated user with an IAM role.
2. In the Amazon S3 console, choose a bucket name to view details about a bucket.
3. Choose the **Permissions** tab.
4. In the **Access control list** section, under **Bucket owner**, the canonical ID for your AWS account appears.

AWS CLI & SDKs

To find the canonical user ID using the AWS CLI

The same AWS CLI and API command works for the AWS account root user, IAM users, or IAM roles.

Use the list-buckets command as follows.

```
$ aws s3api list-buckets \
    --max-items 10 \
    --page-size 10 \
    --query Owner.ID \
    --output text
249fa2f1dc32c330EXAMPLE91b2778fcc65f980f9172f9cb9a5f50ccbEXAMPLE
```

Troubleshoot your AWS account

This information is about troubleshooting your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

Use the information in the following topics to help you diagnose and fix issues with your AWS account. For help with the root user, see Troubleshooting issues with the root user in the *IAM User Guide*. For help with the sign-in process, see Troubleshooting AWS account sign-in issues in the *AWS Sign-In User Guide*.

Troubleshooting topics

- Troubleshooting issues with AWS account creation
- Troubleshooting issues with AWS account closure
- Troubleshooting other issues with AWS accounts

Troubleshooting issues with AWS account creation

This information is about troubleshooting issues with AWS account creation if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

Use the reference links in the following table to help you diagnose and fix issues with creating a new AWS account.

Issue	Reference link	Source
I don't know how to sign-up or create an account	Sign up for AWS (advanced)	This guide
What should I do if I didn't receive a call from AWS to verify my new account or the PIN I entered doesn't work?	https://repost.aws/knowledge-center/phone-verify-no-call	AWS re:Post
How do I resolve the "maximum number of failed attempts" error when I try to verify my AWS account by phone?	https://repost.aws/knowledge-center/maximum-failed-attempts	AWS re:Post
It's been more than 24 hours and my account isn't activated	https://repost.aws/knowledge-center/create-and-activate-aws-account	AWS re:Post
I can't sign into my new account after it has been created	https://docs.aws.amazon.com/signin/latest/userguide/troubleshooting-sign-in-issues.html	AWS Sign-In User Guide

For additional help, we recommend that you search AWS re:Post for content related to your specific issue. If you still need assistance, contact AWS Support.

Troubleshooting issues with AWS account closure

This information is about troubleshooting issues with AWS account closure if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

Use the information below to help you diagnose and fix common issues found during the account closure process. For general information about the account closure process, see Close an AWS account.

Topics

I don’t know how to delete or cancel my account

To close your account, follow the instructions in Close an AWS account.

I don’t see the Close account button on the Accounts page

If you are not signed in as the root user, you will not see the **Close account** button displayed on the **Accounts** page. You must Sign in to the AWS Management Console as the root user to close your account. If you can’t sign in, see Troubleshooting issues with the root user.

I closed my account but still haven’t received an email confirmation

This confirmation email is only sent to the root user email address for the AWS account. If you don’t receive this email within a few hours, you can Sign in to the AWS Management Console as the root user to check that your account is closed. If your account was closed successfully, you will see a message displayed that indicates your account is closed. If the account you closed is a member account, you can verify successful closure by checking if the closed account is labeled as `CLOSED` in the AWS Organizations console. For more information, see Closing a member account in your organization in the *AWS Organizations User Guide*.

If you are trying to close a **management account** and do not receive an email confirmation about the account closure, your organization most likely has active member accounts. You can only close the management account if your organization doesn’t have any active member accounts. To verify that there are no active member accounts remaining in your organization, go to the AWS Organizations console, and make sure that all member accounts are showing `Closed` next to their account names. After that, you can close the management account.

I receive a "ConstraintViolationException" error when trying to close my account

You are trying to close a management account using the AWS Organizations console, which is not possible. To close a management account, you need to Sign in to the AWS Management Console as the root user for the management account and close it from the **Accounts** page. For more information, see Closing a management account in your organization in the *AWS Organizations User Guide*.

I receive a "CLOSE_ACCOUNT_QUOTA_EXCEEDED" error when trying to close a member account

You can only close 20% of member accounts within a rolling 30 day period. This quota is not bound by a calendar month, but starts when you close an account. Within 30 days of that initial account closure, you can't exceed the 20% account closure limit. The minimum account closure is 250 and the maximum account closure is 1000, even if 20% of accounts exceeds 1000. For more information about Organizations quotas, see Quotas for AWS Organizations in the *AWS Organizations User Guide*.

Do I need to delete my AWS organization before closing the management account?

No, you don’t need to delete your AWS organization before closing the management account. However, you can only close the management account if your organization doesn’t have any active member accounts. To verify that there are no active member accounts remaining in your organization, go to the AWS Organizations console, and make sure that all member accounts are showing `Closed` next to their account names. After that, you can close the management account.

Troubleshooting other issues with AWS accounts

This information is about troubleshooting other issues with your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

Use the information here to help you troubleshoot issues related to your AWS account.

Issues

- I need to change the credit card for my AWS account
- I need to report fraudulent AWS account activity
- I need to close my AWS account

I need to change the credit card for my AWS account

To change the credit card for your AWS account, you must be able to sign in. AWS has protections in place that require you to prove that you are the account owner. For instructions, see Managing your credit card payment methods in the *AWS Billing User Guide*.

I need to report fraudulent AWS account activity

If you suspect fraudulent activity using your AWS account and would like to make a report, see How do I report abuse of AWS resources.

If you are having trouble with a purchase made on Amazon.com, see Amazon Customer Service.

I need to close my AWS account

For help troubleshooting issues with closing your AWS account, see Close an AWS account.

Manage accounts in India

This information is about managing accounts in India if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

If you sign up for a new AWS account and choose India for your contact and billing address, your user agreement is with Amazon Web Services India Private Limited (AWS India), a local AWS seller in India. AWS India manages your billing, and your invoice total is listed in Indian rupees (INR) instead of US dollars (USD). For information about managing an AWS account, see Configure your AWS account.

If your account is with AWS India, follow the procedures in this topic to manage your account. This topic explains how to sign up for an AWS India account, edit information about your AWS India account, manage customer verification, and add or edit your Permanent Account Number (PAN).

As part of the credit card verification during sign up, AWS India charges your credit card 2 INR. AWS India refunds the 2 INR after verification is done. You might be redirected to your bank as part of the verification process.

Topics

- Create an AWS account with AWS India
- Manage your customer verification information

Create an AWS account with AWS India

AWS India is a local seller of AWS in India. If your contact and billing address is in India and you want to create an account, use the following procedure to sign up for an AWS India account.

To sign up for an AWS India account

1. Open the Amazon Web Services home page.
2. Choose **Create an AWS account**.

   > **Note**
   >
   > If you signed in to AWS recently, that option might not be there. Instead, choose **Sign in to the Console**. If **Create a new AWS account** still isn't visible, choose **Sign in to a different account**, and then choose **Create a new AWS account**.

3. Enter your account information, verify your email address, and choose a strong password for your account.
4. Choose **Business** or **Personal**. Personal accounts and business accounts have the same features and functions.
5. Enter your company or personal contact information. If your contact or billing address is based in India, in compliance with Indian Computer Emergency Response Team (CERT-In) regulations, AWS is required to collect and validate your identity information before granting you access to AWS services.

   The name that you choose between your contact or billing information must exactly match the name that appears on the document you plan to use for customer verification. For example, if you plan to verify a business account using a Certificate of Incorporation, you must provide the business name that appears on the document. For a list of accepted document types, see the section called “Accepted India documents for customer verification”.
6. After you have read the customer agreement, select the terms and conditions check box, and then choose **Continue**.
7. On the **Billing information** page, enter the payment method that you want to use. You must provide your CVV as part of the verification process.
8. Under **Do you have a PAN?**, choose **Yes** if you have a Permanent Account Number (PAN) that you would like to be displayed on your tax invoices, and then enter your PAN. If you do not have a PAN or want to add it after sign up, choose **No**.

9. Choose **Verify and continue**. AWS India charges your card 2 INR as part of the verification process. AWS India refunds the 2 INR after verification is done.
10. On the **Confirm your identity** page, select your primary purpose of account registration.
11. Choose the ownership type that best represents the owner of the account. If you choose a company, organization, or partnership as the ownership type, enter the name of a key managerial person. The key managerial person can be a director, head of operations, or a person in charge of operations in your business.
12. Depending on the ownership type you chose, choose an accepted India document type to use for verification and type your document information.

    > **Note**
    >
    > If you have a personal account and plan to use a driving license that is *not* issued by the Union of India, we recommend using a different personal document type for verification.

13. Choose the name you want to use for customer verification.

    The names from your billing and contact information will appear for selection if they are associated with an Indian address. Make sure the name you choose matches the name on the document type you plan to use for customer verification. If you need to make changes to the name associated with your billing or contact address, you can do so after you complete account sign up.
14. Provide your consent to submit the information for verification, and then choose **Continue**.

    You will be notified of the customer verification result by email after you complete account signup. You can also check the status on the **Customer verification** page in your account settings or in the AWS Health Dashboard later. You must pass customer verification to access AWS services.
15. Choose whether you want to verify your mobile phone number by **Text message (SMS)** or **Voice call**.
16. Select your country or region code, and then enter your mobile phone number.
17. Complete the security check.
18. Choose **Send SMS** or **Call Me Now**. After a few moments, you will receive a four-digit pin in an SMS or an automated call on your mobile phone.
19. On the **Confirm your identity** page, enter the pin you received and choose **Continue**.

20. On the **Select a support plan** page, select your support plan, and then choose **Complete sign up**. After your payment method and customer verification are verified, your account will be activated and you will receive an email confirming the activation of your account.

> **Note**
>
> If you have completed customer verification and you edit the name, address, or document previously used to verify your identity, you may need to update and complete your customer verification again. For more information, see the section called "Edit your customer verification information".

Manage your customer verification information

In compliance with Indian Computer Emergency Response Team (CERT-In) regulations, AWS is required to collect and validate your identity information before granting you new or continued access to AWS services. Your identity must be verified using the name from the India billing or contact address you provided. During verification, AWS will check if the document number is valid and if the name you provide matches the name associated with the document you use for customer verification. The name that you choose between your contact or billing information must exactly match the name that appears on the document.

To update your billing name and address, see the Payment preferences page. To update your contact name and address, see the section called "Update the primary contact for your AWS account". If you edit any information you previously used for customer verification, such as the name or India-based address from your billing or contact information, you may need to update and re-submit your customer verification information.

Check your customer verification status

You can view your customer verification status at any time on the **Customer verification** page. If your verification status is **Verification required** or **Verification failed**, create or update your customer verification information and submit it for verification.

If your management account has enabled customer verification inheritance, your account's verification status reflects that account's status. You do not need to take any action when inheritance is active.

Create your customer verification information

To complete customer verification, you will need to provide information from an accepted India document. For a list of accepted document types, see the section called "Accepted India documents for customer verification".

1. Sign into the AWS Management Console.
2. On the navigation bar, in the upper-right corner, choose your account name (or alias), and then choose **Account**.
3. Under **Other settings**, choose **Customer verification**.

   If you have not yet provided your customer verification information before, you will see the **Create customer verification** page.
4. Choose the name that exactly matches the name on the document you plan to use for customer verification. For example, if you plan to verify a business account using a Certificate of Incorporation, you must provide the business name that appears on the document.
5. Provide the remaining information requested on the page. Depending on the document type you chose, you may need to upload a copy of both the front and back of the document. If you upload an image file, make sure all the information in the document is visible and legible.
6. Choose **Submit**.

   You will be notified of the customer verification result and any next steps by email or on the AWS Health Dashboard.

Edit your customer verification information

You can edit your customer verification information, such as your primary purpose of account registration, your organization type, and the name, document type, document upload, or document information you want to use for verification.

If you edit the name or document type to use for customer verification, or update any document information, you must verify your identity again when you save the changes.

Verification inheritance is active

If your management account has enabled customer verification inheritance, you cannot edit or submit your own customer verification information. Your verification information appears in read-only mode.

1. Sign into the AWS Management Console.
2. On the navigation bar, in the upper-right corner, choose your account name (or alias), and then choose **Account**.
3. Under **Other settings**, choose **Customer verification**.
4. Choose **Edit**, and then update the information you want to change.

   As you update the information, note the following guidance:

   - If you choose a different name, the name must exactly match the name on the document you plan to use for customer verification. For example, if you plan to verify a business account using a Certificate of Incorporation, you must provide the business name that appears on the document.
   - If you choose a different document type, you will need to upload a copy of the front and back (if applicable) of the document. All the information in the document upload should be visible and legible.
   - If you have a personal account and plan to use a driving license that is *not* issued by the Union of India, we recommend using a different personal document type for verification.

     For a list of accepted document types, see the section called "Accepted India documents for customer verification".
5. Choose **Submit**.

   If your identity must be verified again due to the type of changes you saved, you will be notified of the customer verification result and any next steps by email. You can also view the results by returning to the **Customer verification** page or in the AWS Health Dashboard.

Manage customer verification inheritance for AWS Organizations

With AWS Organizations, you can enable customer verification inheritance for your organization. When you enable inheritance, India-based member accounts in your organization inherit your

management account's verification status. They do not need to complete customer verification independently.

Your management account must complete its own customer verification before it can enable inheritance. After you enable inheritance, you do not need to resubmit your verification documents.

When your management account enables customer verification inheritance, your verification status extends to India-based member accounts in your organization. For those member accounts, we might use or disclose your verification information in response to law enforcement or regulatory requests.

Enable or disable customer verification inheritance

Use the following procedure to turn customer verification inheritance on or off for your organization's India-based member accounts.

1. Open the AWS Management Console at AWS Management Console.
2. On the navigation bar, choose your account name (or alias), and then choose **Account**.
3. Under **Other settings**, choose **Customer verification**.
4. Under **Customer verification settings**, turn the **Inheritance** toggle on to enable inheritance or off to disable it.
5. Review the confirmation dialog box, and then confirm your choice.

When you enable inheritance, all India-based member accounts in your organization inherit your verification status. When you disable inheritance, we restore the prior verification status for member accounts that previously completed their own verification. A member account without prior verification must complete customer verification independently.

Customer verification for AWS Organizations member accounts

If your account has an India-based contact or billing address, you inherit the management account's verification status. This applies when your management account has enabled customer verification inheritance. You can see a message on the **Customer verification** page that confirms this.

When inheritance is active, you cannot submit or edit your own customer verification information. You do not need to take any action.

If your management account disables inheritance, you must have your own customer verification status. The same applies if your account leaves the organization, or the management account removes your account from the organization. If you previously submitted your own verification information, we restore your prior verification status. You do not need to repeat the process. This restoration applies only if your name and address have not changed since your last verification.

Accepted India documents for customer verification

The following document types issued by the Indian government are accepted for customer verification.

> **Note**
>
> The following links are subject to change by the government.

- **PAN Card** - Available in both digital and physical formats, the Permanent Account Number (PAN) card contains a unique alphanumeric identifier issued by the Income Tax Department of India to individuals, companies, and entities. A PAN consists of ten characters, including letters and numbers, in the format **AAAAA1111A**. To use this document for verification, you must also provide the date of birth (individual) or date of incorporation (business) that appears on the PAN document and upload the front side of the card. You can go to the Income Tax Department's official website to check the validity of your PAN.
- **Voter ID Card/EPIC** - The voter ID card, also known as the Electors Photo Identity Card (EPIC), contains a unique identification number issued by Election Commission of India to eligible voters in India. A voter ID/EPIC number consists of ten characters, including letters and numbers. You can go to the official website of the Election Commission of India to check the validity of your voter ID. To use this document for verification, you must upload both the front and back side of the card.
- **Driving License** - If your driving license is *not* issued by the Union of India, we recommend using a different document type for verification. A driving license number consists of 12-16 characters, including letters, numbers, and a space or hyphen. To use this document for verification, you must provide your date of birth and upload both the front and back side of the card. You can go to the Parivahan Sewa site of the Ministry of Road Transport and Highways to check the validity of your driving license.
- **Certificate of Incorporation** - A certificate of incorporation is a document issued by the Ministry of Corporate Affairs (MCA) that dates the registration of a business as a legal entity. The

certificate is used to uniquely identify and track companies registered in India. Each certificate contains a Corporate Identification Number (CIN), which is a unique alphanumeric identifier consisting of 21 characters, including letters and numbers. To use this document for verification, you must upload the certificate of incorporation document. You can go to the Ministry of Corporate Affairs portal to check the validity of your CIN.

Different India document types are accepted for personal and business accounts:

- For personal accounts - PAN card, voter ID card/EPIC, and driving license.
- For business accounts - PAN card and certificate of incorporation.

Manage your AWS India account

Except for the following tasks, the procedures for managing your account are the same as accounts created outside of India. For general information about managing your account, see the section called “Configure your account”.

Use the AWS Management Console to perform the following tasks:

- Adding or editing a Permanent Account Number
- Editing multiple Permanent Account Numbers
- the section called “Manage your customer verification information”
- Edit multiple Goods and Services Tax Numbers (GSTs)
- View a tax invoice

Close an AWS account

These instructions are for how to close your AWS account if you signed up for AWS using Sign up for AWS (advanced) or if you activated advanced features for your account. To compare sign-up options, see Compare sign-up options.

If you no longer need your AWS account, you can close it at any time by following the instructions in this section. After you've closed it, you can reopen it within 90 days from the day you closed the account. The timespan between the day you closed the account and when AWS permanently closes the account is referred to as the post-closure period.

What you need to know before closing your account

Before closing your AWS account, you should consider the following:

- Closing your account will serve as your notice of termination of the AWS Customer Agreement for this account.
- You don't need to delete resources in your AWS account before closing it. However, we recommend you back up any resources or data that you want to keep. For instructions about how to back up a particular resource, see the appropriate AWS documentation for that service.
- You can reopen your account during the post-closure period. Charges for the services that remained in your account will restart if you reopen it. You also remain responsible for any unpaid invoices and outstanding Reserved Instances and Savings Plans.
- You remain responsible for all outstanding fees and charges for the services consumed before account closure. You will receive an AWS bill the following month after closing your account. For example, if you closed your account on January 15, you will receive a bill at the beginning of February for usage incurred from January 1 through January 15. You will continue receiving invoices for Reserved Instances and Savings Plans after closing your account until they expire.
- You will no longer be able to access AWS services that were previously available in your account. However, you can sign in and access a closed AWS account during the post-closure period only to view past billing information, access account settings, or contact AWS Support.
- You can't use the same email address that was registered to your AWS account at the time of its closure as the primary email of another AWS account. If you want to use the same email address for a different AWS account, we recommend updating it before closure. For more information, see Update the root user email address.
- You can't use the same account alias that you registered to your AWS account at the time of its closure for another AWS account. To reuse the alias for a different AWS account, delete the account alias before you close the account. For more information, see Using an alias for your AWS account ID in the *IAM User Guide*.
- If you've enabled multi-factor authentication (MFA) on your AWS account root user, or configured an MFA device on an IAM user, MFA isn't removed automatically when you close the account. If you choose to leave MFA turned on during the 90 days post-closure period, keep the MFA device active until the post-closure period has expired in case you need to access the account during that time. Note, the hardware TOTP token devices cannot be associated with another user after the permanent closure of your account. If you would like to use the hardware TOTP token with another user later, you have the option to deactivate the hardware MFA device before closing the account. MFA devices for IAM users must be deleted by the account administrator.

Additional considerations for member accounts

- When you close a member account, that account isn't removed from the organization until after the post-closure period. During the post-closure period, a closed member account still counts toward your quota of accounts in the organization. To avoid having the account count against the quota, see Remove a member account from your organization before closing it.
- You can only close 20% or 250 of member accounts to a maximum of 1,000 within a rolling 30 day period, whichever is higher. This quota is not bound by a calendar month, but starts when you close an account. For more information about Organizations quotas, see Quotas for AWS Organizations.
- If you use AWS Control Tower, you need to unmanage the member account before you attempt to close the account. See Unmanage a member account in the *AWS Control Tower User Guide*.

Service specific considerations

- AWS Marketplace subscriptions aren't automatically canceled on account closure. If you have any subscriptions, first terminate all instances of your software in the subscriptions. Then, go to the **Manage subscriptions** page of the AWS Marketplace console and cancel your subscriptions.
- After an account has been closed, AWS will send daily emails for up to five days before we suspend the domain. After the domain has been suspended, and depending on the domain's registrar, we will either delete the domain within 30 days or release the domain to its registrar. For more information, see My AWS account is closed or permanently closed, and my domain is registered with Route 53.

  If the email address for this account's root user uses a domain registered through Route 53 in this same account, do not close this account until you complete one of the following actions: transfer the domain to a different AWS account, or update the root user email address to use a domain that is not registered in this account. Otherwise, closing the account suspends the domain, which breaks email delivery to your root user address. Without email, you cannot recover the account, and without the account, you cannot reinstate the domain. After 30 days, Route 53 permanently deletes the domain. This also applies to member accounts whose root user email addresses use domains registered in this account.
- AWS CloudTrail is a foundational security service. This means that trails created by users can continue to exist and deliver events even after an AWS account is closed, unless a user explicitly deletes the trails in their AWS account before closing it. For more information about how to

request trail deletion after an AWS account has been closed, see AWS account closure and trails in the *CloudTrail User Guide*.

How to close your account

You can close your AWS account using the following procedure. Note, that there is different guidance provided in each tab depending on the type of account [standalone, member, management, and AWS GovCloud (US)] you want to close.

If you experience any issues during the process of closing your account, see Troubleshooting issues with AWS account closure.

Standalone account

A standalone account is an individually managed account that is not part of AWS Organizations.

To close a standalone account from the Accounts page

1. Sign in to the AWS Management Console as the root user in the AWS account that you want to close. You can't close an account while signed in as an IAM user or role.
2. On the navigation bar in the upper-right corner, choose your account name or number, and then choose **Account**.
3. On the **Account** page, choose the **Close account** button.
4. Type your account ID (displayed at the top of the closure dialog box) to confirm that you have read and understand the account closure process.
5. Choose the **Close account** button to initiate the account closure process.
6. Within a few minutes, you should receive an email confirmation that your account has been closed.

> **Note**
>
> This task isn't supported in the AWS CLI or by an API operation from one of the AWS SDKs. You can perform this task only by using the AWS Management Console.

Member account

A member account is an AWS account that is part of AWS Organizations.

To close a member account from the AWS Organizations console

1. Sign in to the AWS Organizations console.
2. On the **AWS accounts** page, find and choose the name of the member account you want to close. You can navigate the OU hierarchy, or look at a flat list of accounts without the OU structure.
3. Choose **Close** next to the account name at the top of the page. This option is only available when an AWS organization is in All features mode.

   > **Note**
   >
   > If your organization is using Consolidated billing mode, you won't be able to see the **Close** button in the console. To close an account in consolidated billing mode, sign in to the account you want to close as the root user. On the **Accounts** page, choose the **Close account** button, enter your account ID, and then choose the **Close account** button.

4. Read and ensure that you understand the account closure guidance.
5. Enter the member account ID, and then choose **Close account** to initiate the account closure process.

> **Note**
>
> Any member account that you close will display a CLOSED label next to its account name in the AWS Organizations console for up to 90 days after the original closure date. After 90 days, the member account will no longer be displayed in the AWS Organizations console.

To close a member account from the Accounts page

Optionally, you can close an AWS member account directly from the **Account** page in the AWS Management Console. For step-by-step guidance, follow the instructions in the **Standalone account** tab.

To close a member account using AWS CLI and SDKs

For instructions on how to close a member account using the AWS CLI and SDKs, see Closing a member account in your organization in the *AWS Organizations User Guide*.

Management account

A management account is an AWS account that acts as the parent or root account for AWS Organizations.

> **Note**
>
> You cannot close a management account directly from the AWS Organizations console.

To close a management account from the Accounts page

1. Sign in to the AWS Management Console as the root user for the management account that you want to close. You can't close an account while signed in as an IAM user or role.
2. Verify that there are no active member accounts remaining in your organization. To do this, go to the AWS Organizations console, and make sure that all member accounts are showing `Closed` next to their account names. If you have a member account that is still active, you will need to follow the account closure guidance provided in the **Member account** tab before you can move to the next step.
3. On the navigation bar in the upper-right corner, choose your account name or number, and then choose **Account**.
4. On the **Account** page, choose the **Close account** button.
5. Type your account ID (displayed at the top of the closure dialog box) to confirm that you have read and understand the account closure process.
6. Choose the **Close account** button to initiate the account closure process.
7. Within a few minutes, you should receive an email confirmation that your account has been closed.

> **Note**
>
> This task isn't supported in the AWS CLI or by an API operation from one of the AWS SDKs. You can perform this task only by using the AWS Management Console.

AWS GovCloud (US) account

An AWS GovCloud (US) account is always linked to a single standard AWS account for billing and payment purposes.

To close an AWS GovCloud (US) account

If you have an AWS account that is linked to an AWS GovCloud (US) account, you need to close the standard account before you close the AWS GovCloud (US) account. For more details, including how to back-up data and avoid unintended AWS GovCloud (US) charges, see Closing an AWS GovCloud (US) account in the *AWS GovCloud (US) User Guide*.

What to expect after you close your account

Immediately after you close your account, the following will occur:

- You will receive an email confirming the account closure to the root user's email address. If you don't receive this email within a few hours, see Troubleshooting issues with AWS account closure.
- Any member account that you close will display a CLOSED label next to its account name in the AWS Organizations console for up to 90 days after the original closure date. After 90 days, the member account will no longer be displayed in the AWS Organizations console.
- If you have granted permissions to access services in your AWS account to other accounts, any access requests made from those accounts should fail after account closure. If you reopen your AWS account, other AWS accounts can again access your account's AWS services and resources if you granted the necessary permissions to them.

Account closure may not occur immediately across all Regions and services and can take several hours to complete.

Post-closure period

The post-closure period refers to the length of time between the day you closed your account and when AWS permanently closes your AWS account. The post-closure period is 90 days. During the post-closure period, you can access your content and AWS services only by reopening your account. After the post-closure period, AWS permanently closes your AWS account, and you can no longer reopen it. AWS will also delete content and resources in your account (except for CloudTrail trails). After an account has been permanently closed, its AWS account ID can never be reused.

Reopening your AWS account

Your account will permanently close in 90 days, after which you will not be able to reopen your account and AWS will delete the content remaining in your account. To reopen your account before it is permanently closed, (1) you must contact AWS Support as soon as possible, and (2) we must receive full payment of any outstanding balance, including providing required information as specified on the invoice, within 30 days from the date of account closure.

> **Note**
>
> Charges for the services that remained in your account will restart if you reopen it.

Security in AWS Account Management

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security of the cloud and security in the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. Third-party auditors regularly test and verify the effectiveness of our security as part of the AWS Compliance Programs. To learn about the compliance programs that apply to Account Management, see AWS services in scope by compliance program.
- **Security in the cloud** – Your responsibility is determined by the AWS service that you use. You are also responsible for other factors including the sensitivity of your data, your company's requirements, and applicable laws and regulations.

The shared responsibility model is enforced for all AWS accounts. This includes AWS accounts you create using our new AWS experience and accounts you create using our advanced AWS experience.

This documentation helps you understand how to apply the shared responsibility model when using AWS Account Management. It shows you how to configure Account Management to meet your security and compliance objectives. You also learn how to use other AWS services that help you to monitor and secure your Account Management resources. Some of this documentation is only relevant for accounts created with our advanced AWS experience. We'll identify when documentation is only relevant for accounts created with our advanced AWS experience.

Topics

- Infrastructure security in AWS Account Management

Data protection in AWS Account Management

The following information is relevant for all AWS accounts.

The AWS shared responsibility model applies to data protection in AWS Account Management. As described in this model, AWS is responsible for protecting the global infrastructure that runs all of the AWS Cloud. You are responsible for maintaining control over your content that is hosted on this infrastructure. You are also responsible for the security configuration and management tasks for the AWS services that you use. For more information about data privacy, see Data Privacy FAQ. For information about data protection in Europe, see the General Data Protection Regulation (GDPR) Center.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual users with AWS IAM Identity Center or AWS Identity and Access Management (IAM). That way, each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources. We require TLS 1.2 and recommend TLS 1.3.
- Set up API and user activity logging with AWS CloudTrail. For information about using CloudTrail trails to capture AWS activities, see Working with CloudTrail trails in the *AWS CloudTrail User Guide*.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing sensitive data that is stored in Amazon S3.
- If you require FIPS 140-3 validated cryptographic modules when accessing AWS through a command line interface or an API, use a FIPS endpoint. For more information about the available FIPS endpoints, see Federal Information Processing Standard (FIPS) 140-3.

We strongly recommend that you never put confidential or sensitive information, such as your customers' email addresses, into tags or free-form text fields such as a **Name** field. This includes when you work with Account Management or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into tags or free-form text fields used for names may be used for billing or diagnostic logs. If you provide a URL to an external server, we strongly

recommend that you do not include credentials information in the URL to validate your request to that server.

AWS PrivateLink for AWS Account Management

The following information is relevant for all AWS accounts.

If you use Amazon Virtual Private Cloud (Amazon VPC) to host your AWS resources, you can access the AWS Account Management service from within the VPC without having to cross the public internet.

Amazon VPC lets you launch AWS resources in a custom virtual network. You can use a VPC to control your network settings, such as the IP address range, subnets, route tables, and network gateways. For more information about VPCs, see the *Amazon VPC User Guide*.

To connect your Amazon VPC to Account Management, you must first define an *interface VPC endpoint*, which lets you connect your VPC to other AWS services. The endpoint provides reliable, scalable connectivity, without requiring an internet gateway, network address translation (NAT) instance, or VPN connection. For more information, see Interface VPC Endpoints (AWS PrivateLink) in the *Amazon VPC User Guide*.

Creating the Endpoint

You can create an AWS Account Management endpoint in your VPC using the AWS Management Console, the AWS Command Line Interface (AWS CLI), an AWS SDK, the AWS Account Management API, or CloudFormation.

For information about creating and configuring an endpoint using the Amazon VPC console or the AWS CLI, see Creating an Interface Endpoint in the *Amazon VPC User Guide.*

> **Note**
>
> When you create an endpoint, specify Account Management as the service that you want your VPC to connect to, using the following format:
>
> ```
> com.amazonaws.us-east-1.account
> ```
>
> You must use the string exactly as shown, specifying the `us-east-1` Region. As a global service, Account Management is hosted in only that one AWS Region.

For information about creating and configuring an endpoint using CloudFormation, see the AWS::EC2::VPCEndpoint resource in the *CloudFormation User Guide*.

Amazon VPC Endpoint Policies

You can control what actions can be performed through this service endpoint by attaching an endpoint policy when you create the Amazon VPC endpoint. You can create complex IAM rules by attaching multiple endpoint policies. For more information, see:

- Amazon Virtual Private Cloud endpoint policies for Account Management
- Controlling Access to Services with VPC Endpoints in the *AWS PrivateLink Guide*.

Amazon Virtual Private Cloud endpoint policies for Account Management

The following information is relevant for all AWS accounts.

You can create a Amazon VPC endpoint policy for Account Management in which you specify the following:

- The principal that can perform actions.
- The actions that the principals can perform.
- The resources on which the actions can be performed.

The following example shows an Amazon VPC endpoint policy that allows one IAM user named Alice in account 123456789012 to both retrieve and change the alternate contact information for any AWS account, but denies all IAM users permission to delete any alternate contact information on any account.

If you want to grant access to accounts that are part of an AWS Organization to a principal that is in one of the organization's member accounts, then the `Resource` element must use the following format:

```
arn:aws:account::{ManagementAccountId}:account/o-{OrganizationId}/{AccountId}
```

For more information about creating endpoint policies, see Controlling Access to Services with VPC Endpoints in the *AWS PrivateLink Guide*.

Identity and Access Management for AWS Account Management

AWS Identity and Access Management (IAM) is an AWS service that helps an administrator securely control access to AWS resources. IAM administrators control who can be *authenticated* (signed in) and *authorized* (have permissions) to use Account Management resources. IAM is an AWS service that you can use with no additional charge.

The following information is about Identity and Access Management for AWS Account Management. This information is most relevant for AWS accounts that you create when you use our advanced AWS experience or if you activated advanced features for your account. To learn about how access control works for our new AWS experience, see Compare access management.

Topics

Audience

How you use AWS Identity and Access Management (IAM) differs based on your role:

- **Service user** - request permissions from your administrator if you cannot access features (see Troubleshooting AWS Account Management identity and access)
- **Service administrator** - determine user access and submit permission requests (see How AWS Account Management works with IAM)
- **IAM administrator** - write policies to manage access (see Identity-based policy examples for AWS Account Management)

Authenticating with identities

Authentication is how you sign in to AWS using your identity credentials. You must be authenticated as the AWS account root user, an IAM user, or by assuming an IAM role.

You can sign in as a federated identity using credentials from an identity source like AWS IAM Identity Center (IAM Identity Center), single sign-on authentication, or Google/Facebook credentials. For more information about signing in, see How to sign in to your AWS account in the *AWS Sign-In User Guide*.

For programmatic access, AWS provides an SDK and CLI to cryptographically sign requests. For more information, see AWS Signature Version 4 for API requests in the *IAM User Guide*.

AWS account root user

When you create an AWS account, you begin with one sign-in identity called the AWS account *root user* that has complete access to all AWS services and resources. We strongly recommend that you don't use the root user for everyday tasks. For tasks that require root user credentials, see Tasks that require root user credentials in the *IAM User Guide*.

Federated identity

As a best practice, require human users to use federation with an identity provider to access AWS services using temporary credentials.

A *federated identity* is a user from your enterprise directory, web identity provider, or Directory Service that accesses AWS services using credentials from an identity source. Federated identities assume roles that provide temporary credentials.

For centralized access management, we recommend AWS IAM Identity Center. For more information, see What is IAM Identity Center? in the *AWS IAM Identity Center User Guide*.

IAM users and groups

An *IAM user* is an identity with specific permissions for a single person or application. We recommend using temporary credentials instead of IAM users with long-term credentials. For more information, see Require human users to use federation with an identity provider to access AWS using temporary credentials in the *IAM User Guide*.

An *IAM group* specifies a collection of IAM users and makes permissions easier to manage for large sets of users. For more information, see Use cases for IAM users in the *IAM User Guide*.

IAM roles

An *IAM role* is an identity with specific permissions that provides temporary credentials. You can assume a role by switching from a user to an IAM role (console) or by calling an AWS CLI or AWS API operation. For more information, see Methods to assume a role in the *IAM User Guide*.

IAM roles are useful for federated user access, temporary IAM user permissions, cross-account access, cross-service access, and applications running on Amazon EC2. For more information, see Cross account resource access in IAM in the *IAM User Guide*.

Managing access using policies

You control access in AWS by creating policies and attaching them to AWS identities or resources. A policy defines permissions when associated with an identity or resource. AWS evaluates these policies when a principal makes a request. Most policies are stored in AWS as JSON documents. For more information about JSON policy documents, see Overview of JSON policies in the *IAM User Guide*.

Using policies, administrators specify who has access to what by defining which **principal** can perform **actions** on what **resources**, and under what **conditions**.

By default, users and roles have no permissions. An IAM administrator creates IAM policies and adds them to roles, which users can then assume. IAM policies define permissions regardless of the method used to perform the operation.

Identity-based policies

Identity-based policies are JSON permissions policy documents that you attach to an identity (user, group, or role). These policies control what actions identities can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Define custom IAM permissions with customer managed policies in the *IAM User Guide*.

Identity-based policies can be *inline policies* (embedded directly into a single identity) or *managed policies* (standalone policies attached to multiple identities). To learn how to choose between managed and inline policies, see Choose between managed policies and inline policies in the *IAM User Guide*.

Resource-based policies

Resource-based policies are JSON policy documents that you attach to a resource. Examples include IAM *role trust policies* and Amazon S3 *bucket policies*. In services that support resource-

based policies, service administrators can use them to control access to a specific resource. You must specify a principal in a resource-based policy.

Resource-based policies are inline policies that are located in that service. You can't use AWS managed policies from IAM in a resource-based policy.

Other policy types

AWS supports additional policy types that can set the maximum permissions granted by more common policy types:

- **Permissions boundaries** – Set the maximum permissions that an identity-based policy can grant to an IAM entity. For more information, see Permissions boundaries for IAM entities in the *IAM User Guide*.
- **Service control policies (SCPs)** – Specify the maximum permissions for an organization or organizational unit in AWS Organizations. For more information, see Service control policies in the *AWS Organizations User Guide*.
- **Resource control policies (RCPs)** – Set the maximum available permissions for resources in your accounts. For more information, see Resource control policies (RCPs) in the *AWS Organizations User Guide*.
- **Session policies** – Advanced policies passed as a parameter when creating a temporary session for a role or federated user. For more information, see Session policies in the *IAM User Guide*.

Multiple policy types

When multiple types of policies apply to a request, the resulting permissions are more complicated to understand. To learn how AWS determines whether to allow a request when multiple policy types are involved, see Policy evaluation logic in the *IAM User Guide*.

How AWS Account Management works with IAM

This information is most relevant for AWS accounts that you create when you use our advanced AWS experience. To learn about how access control works for our new AWS experience, see Compare access management.

Before you use IAM to manage access to Account Management, learn what IAM features are available to use with Account Management.

IAM features you can use with AWS Account Management

IAM feature	Account Management support
Identity-based policies	Yes
Resource-based policies	No
Policy actions	Yes
Policy resources	Yes
Policy condition keys	Yes
ACLs	No
ABAC (tags in policies)	No
Temporary credentials	Yes
Principal permissions	Yes
Service roles	No
Service-linked roles	No

To get a high-level view of how Account Management and other AWS services work with most IAM features, see AWS services that work with IAM in the *IAM User Guide*.

Identity-based policies for Account Management

Supports identity-based policies: Yes

Identity-based policies are JSON permissions policy documents that you can attach to an identity, such as an IAM user, group of users, or role. These policies control what actions users and roles can perform, on which resources, and under what conditions. To learn how to create an identity-based policy, see Define custom IAM permissions with customer managed policies in the *IAM User Guide*.

With IAM identity-based policies, you can specify allowed or denied actions and resources as well as the conditions under which actions are allowed or denied. To learn about all of the elements that you can use in a JSON policy, see IAM JSON policy elements reference in the *IAM User Guide*.

Identity-based policy examples for Account Management

To view examples of Account Management identity-based policies, see Identity-based policy examples for AWS Account Management.

Resource-based policies within Account Management

Supports resource-based policies: No

Resource-based policies are JSON policy documents that you attach to a resource. Examples of resource-based policies are IAM *role trust policies* and Amazon S3 *bucket policies*. In services that support resource-based policies, service administrators can use them to control access to a specific resource. For the resource where the policy is attached, the policy defines what actions a specified principal can perform on that resource and under what conditions. You must specify a principal in a resource-based policy. Principals can include accounts, users, roles, federated users, or AWS services.

To enable cross-account access, you can specify an entire account or IAM entities in another account as the principal in a resource-based policy. For more information, see Cross account resource access in IAM in the *IAM User Guide*.

Policy actions for Account Management

Supports policy actions: Yes

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Action` element of a JSON policy describes the actions that you can use to allow or deny access in a policy. Include actions in a policy to grant permissions to perform the associated operation.

To see a list of Account Management actions, see Actions defined by AWS Account Management in the *Service Authorization Reference*.

Policy actions in Account Management use the following prefix before the action.

```
account
```

To specify multiple actions in a single statement, separate them with commas.

```
"Action": [
      "account:action1",
      "account:action2"
         ]
```

You can specify multiple actions using wildcards (*). For example, to specify all actions that work with an AWS account's alternate contacts, include the following action.

```
"Action": "account:*AlternateContact"
```

To view examples of Account Management identity-based policies, see Identity-based policy examples for AWS Account Management.

Policy resources for Account Management

Supports policy resources: Yes

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Resource` JSON policy element specifies the object or objects to which the action applies. As a best practice, specify a resource using its Amazon Resource Name (ARN). For actions that don't support resource-level permissions, use a wildcard (*) to indicate that the statement applies to all resources.

```
"Resource": "*"
```

The Account Management service supports the following specific resource types in an IAM policy's `Resource` element to help you filter the policy and distinguish between these types of AWS accounts:

- **account**

  This `resource` type matches only standalone AWS accounts that are not member accounts in an organization managed by the AWS Organizations service.
- **accountInOrganization**

This `resource` type matches only AWS accounts that are member accounts in an organization managed by the AWS Organizations service.

To see a list of Account Management resource types and their ARNs, see Resources defined by AWS Account Management in the *Service Authorization Reference*. To learn with which actions you can specify the ARN of each resource, see Actions defined by AWS Account Management.

To view examples of Account Management identity-based policies, see Identity-based policy examples for AWS Account Management.

Policy condition keys for Account Management

Supports service-specific policy condition keys: Yes

Administrators can use AWS JSON policies to specify who has access to what. That is, which **principal** can perform **actions** on what **resources**, and under what **conditions**.

The `Condition` element specifies when statements execute based on defined criteria. You can create conditional expressions that use condition operators, such as equals or less than, to match the condition in the policy with values in the request. To see all AWS global condition keys, see AWS global condition context keys in the *IAM User Guide*.

The Account Management service supports the following condition keys that you can use to provide fine-grained filtering for your IAM policies:

- **account:TargetRegion**

  This condition key takes an argument that consists of a list of AWS Region codes. It lets you filter the policy to affect only those actions that apply to the specified Regions.
- **account:AlternateContactTypes**

  This condition key takes a list of alternate contact types:
  - BILLING
  - OPERATIONS
  - SECURITY

  Using this key lets you filter the request to only those actions that target the specified alternate contact types.

- **account:AccountResourceOrgPaths**

  This condition key takes an argument that consists of a list of paths through your organization's hierarchy to specific organizational units (OU). It lets you filter the policy to affect only target accounts in a matching OU.

  ```
  o-aa111bb222/r-a1b2/ou-a1b2-f6g7h111/*
  ```

- **account:AccountResourceOrgTags**

  This condition key takes an argument that consists of a list of tag keys and values. It lets you filter the policy to affect only those accounts that are members of an organization and that are tagged with the specified tag keys and values.

- **account:EmailTargetDomain**

  This condition key takes an argument that consists of a list of email domains. It lets you filter the policy to affect only those actions that match the specified email domains. This condition key is case-sensitive. You should use `StringEqualsIgnoreCase` instead of `StringEquals` in the condition block of the policy to control the action based on the target email address domain. Here is a sample policy allowing the `account:StartPrimaryEmailUpdate` action to complete when the email domain contains `example.com`, `company.org`, or any combination of case, such as `EXAMPLE.COM`.

  ```
  {
      "Version": "2012-10-17",
      "Statement": [
          {
              "Sid": "AllowConditionKey",
              "Effect": "Allow",
              "Action": [
                  "account:StartPrimaryEmailUpdate"
              ],
              "Resource": "*",
              "Condition": {
                  "StringEqualsIgnoreCase": {
                      "account:EmailTargetDomain": [
                          "example.com",
                          "company.org"
                      ]
                  }
              }
  ```

```
        }
    ]
}
```

To see a list of Account Management condition keys, see Condition keys for AWS Account Management in the *Service Authorization Reference*. To learn with which actions and resources you can use a condition key, see Actions defined by AWS Account Management.

To view examples of Account Management identity-based policies, see Identity-based policy examples for AWS Account Management.

Access control lists in Account Management

Supports ACLs: No

Access control lists (ACLs) control which principals (account members, users, or roles) have permissions to access a resource. ACLs are similar to resource-based policies, although they do not use the JSON policy document format.

Attribute-based access control with Account Management

Supports ABAC (tags in policies): No

Attribute-based access control (ABAC) is an authorization strategy that defines permissions based on attributes. In AWS, these attributes are called *tags*. You can attach tags to IAM entities (users or roles) and to many AWS resources. Tagging entities and resources is the first step of ABAC. Then you design ABAC policies to allow operations when the principal's tag matches the tag on the resource that they are trying to access.

ABAC is helpful in environments that are growing rapidly and helps with situations where policy management becomes cumbersome.

For AWS Account Management, tag-based access control is supported only through the `account:AccountResourceOrgTags/key-name` condition key. The standard `aws:ResourceTag/key-name` condition key is not supported for APIs in the account namespace.

Example JSON policy using the supported condition key

The following example policy allows access to view contact information for accounts tagged with the key "CostCenter" and either value "12345" or "67890" in your organization.

JSON

```
{
   "Version":"2012-10-17",
   "Statement":[
      {
         "Effect":"Allow",
         "Action":[
            "account:GetContactInformation",
            "account:GetAlternateContact"
         ],
         "Resource":"*",
         "Condition":{
            "ForAnyValue:StringEquals":{
               "account:AccountResourceOrgTags/CostCenter":[
                  "12345",
                  "67890"
               ]
            }
         }
      }
   ]
}
```

For more information about ABAC, see Define permissions based on attributes with ABAC authorization and IAM tutorial: Define permissions to access AWS resources based on tags in the *IAM User Guide*.

Using temporary credentials with Account Management

Supports temporary credentials: Yes

Temporary credentials provide short-term access to AWS resources and are automatically created when you use federation or switch roles. AWS recommends that you dynamically generate temporary credentials instead of using long-term access keys. For more information, see Temporary security credentials in IAM and AWS services that work with IAM in the *IAM User Guide*.

Cross-service principal permissions for Account Management

Supports forward access sessions (FAS): Yes

Forward access sessions (FAS) use the permissions of the principal calling an AWS service, combined with the requesting AWS service to make requests to downstream services. For policy details when making FAS requests, see Forward access sessions.

Service roles for Account Management

Supports service roles: No

A service role is an IAM role that a service assumes to perform actions on your behalf. An IAM administrator can create, modify, and delete a service role from within IAM. For more information, see Create a role to delegate permissions to an AWS service in the *IAM User Guide*.

Service-linked roles for Account Management

Supports service-linked roles: No

A service-linked role is a type of service role that is linked to an AWS service. The service can assume the role to perform an action on your behalf. Service-linked roles appear in your AWS account and are owned by the service. An IAM administrator can view, but not edit the permissions for service-linked roles.

For details about creating or managing service-linked roles, see AWS services that work with IAM. Find a service in the table that includes a `Yes` in the **Service-linked role** column. Choose the **Yes** link to view the service-linked role documentation for that service.

Identity-based policy examples for AWS Account Management

This information is most relevant for AWS accounts that you create when you use our advanced AWS experience. To learn about how access control works for our new AWS experience, see Compare access management.

By default, users and roles don't have permission to create or modify Account Management resources. To grant users permission to perform actions on the resources that they need, an IAM administrator can create IAM policies.

To learn how to create an IAM identity-based policy by using these example JSON policy documents, see Create IAM policies (console) in the *IAM User Guide*.

For details about actions and resource types defined by Account Management, including the format of the ARNs for each of the resource types, see Actions, resources, and condition keys for AWS Account Management in the *Service Authorization Reference*.

Topics

Policy best practices

Identity-based policies determine whether someone can create, access, or delete Account Management resources in your account. These actions can incur costs for your AWS account. When you create or edit identity-based policies, follow these guidelines and recommendations:

- **Get started with AWS managed policies and move toward least-privilege permissions** – To get started granting permissions to your users and workloads, use the *AWS managed policies* that grant permissions for many common use cases. They are available in your AWS account. We recommend that you reduce permissions further by defining AWS customer managed policies that are specific to your use cases. For more information, see AWS managed policies or AWS managed policies for job functions in the *IAM User Guide*.
- **Apply least-privilege permissions** – When you set permissions with IAM policies, grant only the permissions required to perform a task. You do this by defining the actions that can be taken on specific resources under specific conditions, also known as *least-privilege permissions*. For more information about using IAM to apply permissions, see Policies and permissions in IAM in the *IAM User Guide*.
- **Use conditions in IAM policies to further restrict access** – You can add a condition to your policies to limit access to actions and resources. For example, you can write a policy condition to specify that all requests must be sent using SSL. You can also use conditions to grant access to service actions if they are used through a specific AWS service, such as CloudFormation. For more information, see IAM JSON policy elements: Condition in the *IAM User Guide*.
- **Use IAM Access Analyzer to validate your IAM policies to ensure secure and functional permissions** – IAM Access Analyzer validates new and existing policies so that the policies adhere to the IAM policy language (JSON) and IAM best practices. IAM Access Analyzer provides more than 100 policy checks and actionable recommendations to help you author secure and functional policies. For more information, see Validate policies with IAM Access Analyzer in the *IAM User Guide*.

- **Require multi-factor authentication (MFA)** – If you have a scenario that requires IAM users or a root user in your AWS account, turn on MFA for additional security. To require MFA when API operations are called, add MFA conditions to your policies. For more information, see Secure API access with MFA in the *IAM User Guide*.

For more information about best practices in IAM, see Security best practices in IAM in the *IAM User Guide*.

Using the Account page in the AWS Management Console

To access the **Account** page in the AWS Management Console, you must have a minimum set of permissions. These permissions must allow you to list and view details about your AWS account. If you create an identity-based policy that is more restrictive than the minimum required permissions, the console won't function as intended for entities (IAM users or roles) with that policy.

To ensure that users and roles can use the Account Management console, you can choose to attach either the `AWSAccountManagementReadOnlyAccess` or `AWSAccountManagementFullAccess` AWS managed policy to the entities. For more information, see Adding permissions to a user in the *IAM User Guide*.

You don't need to allow minimum console permissions for users that are making calls only to the AWS CLI or the AWS API. Instead, in many cases you can choose to allow access to only the actions that match the API operations that you are trying to perform.

Providing read-only access to the Account page in the AWS Management Console

In the following example, you want to grant an IAM user in your AWS account read-only access to the Account page in the AWS Management Console. Users with this policy attached can't make any changes.

The `account:GetAccountInformation` action grants access to view most of the settings on the Account page. However, to view the currently enabled AWS Regions, you must also include the `account:ListRegions` action.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
```

```
        {
            "Sid": "GrantReadOnlyAccessToAccountSettings",
            "Effect": "Allow",
            "Action": [
                "account:GetAccountInformation",
                "account:ListRegions"
            ],
            "Resource": "*"
        }
    ]
}
```

Providing full access to the Account page in the AWS Management Console

In the following example, you want to grant an IAM user in your AWS account full access to the Account page in the AWS Management Console. Users with this policy attached can alter settings for the account.

This example policy builds on the preceding example policy by adding each of the available write permissions (with the exception of CloseAccount), which allows the user to change most of the settings for the account, including the `account:EnableRegion` and `account:DisableRegion` permissions.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Sid": "GrantFullAccessToAccountSettings",
            "Effect": "Allow",
            "Action": [
                "account:GetAccountInformation",
                "account:ListRegions",
                "account:PutContactInformation",
                "account:PutAlternateContact",
                "account:DeleteAlternateContact",
                "account:EnableRegion",
                "account:DisableRegion"
            ],
            "Resource": "*"
```

```
        }
    ]
}
```

Using identity-based policies (IAM policies) for AWS Account Management

This information is most relevant for AWS accounts that you create when you use our advanced AWS experience. To learn about how access control works for our new AWS experience, see Compare access management.

For a full discussion of AWS accounts and IAM users, see What Is IAM? in the *IAM User Guide*.

For instructions on how you can update customer managed policies, see Edit IAM policies in the *IAM User Guide*.

AWS Account Management actions policies

This table summarizes the permissions that grant access to your account settings. For examples of policies that use these permissions, see Identity-based policy examples for AWS Account Management.

> **Note**
>
> To grant IAM users write access to a specific account setting in the **Account** page of the AWS Management Console, you must allow the `GetAccountInformation` permission, in addition to the permission (or permissions) that you want to use to modify that setting.

Permission name	Access level	Description
`account:ListRegions`	List	Grants permission to list the available Regions.
`account:GetAccount Information`	Read	Grants permission to retrieve the account information for an account.

Permission name	Access level	Description
account:GetAlternateContact	Read	Grants permission to retrieve the alternate contacts for an account.
account:GetContactInformation	Read	Grants permission to retrieve the primary contact information for an account.
account:GetPrimaryEmail	Read	Grants permission to retrieve the primary email address of an account.
account:GetRegionOptStatus	Read	Grants permission to get the opt-in status of a Region.
account:AcceptPrimaryEmailUpdate	Write	Grants permission to accept the primary email address update of the member account in an AWS organization.
account:CloseAccount	Write	Grants permission to close an account. **Note** This is a permission for the console only. No API access is available for this permission.
account:DeleteAlternateContact	Write	Grants permission to delete the alternate contacts for an account.

Permission name	Access level	Description
account:DisableRegion	Write	Grants permission to disable use of a Region.
account:EnableRegion	Write	Grants permission to enable use of a Region.
account:PutAccountName	Write	Grants permission to update the name for an account.
account:PutAlternateContact	Write	Grants permission to modify the alternate contacts for an account.
account:PutContactInformation	Write	Grants permission to update the primary contact information for an account.
account:StartPrimaryEmailUpdate	Write	Grants permission to initiate the primary email address update of the member account in an AWS organization.

Troubleshooting AWS Account Management identity and access

This information is most relevant for AWS accounts that you create when you use our advanced AWS experience. To learn about how access control works for our new AWS experience, see Compare access management.

Use the following information to help you diagnose and fix common issues that you might encounter when working with Account Management and IAM.

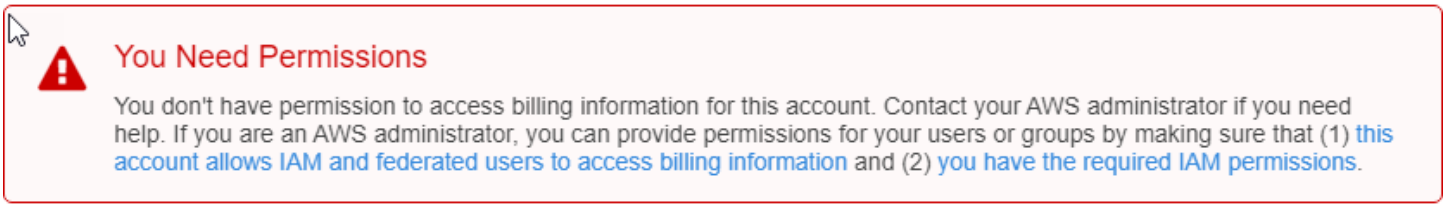
Topics

- I am not authorized to perform an action in the Account page
- I am not authorized to perform iam:PassRole
- I want to allow people outside of my AWS account to access my account details

I am not authorized to perform an action in the Account page

If the AWS Management Console tells you that you are not authorized to perform an action, then you must contact your administrator for assistance. Your administrator is the person that provided you with your user name and password.

The following example error occurs when the `mateojackson` IAM user tries to use the console to view details about his AWS account in the **Account** page of the AWS Management Console but doesn't have the `account:GetAccountInformation` permissions.



In this case, Mateo asks his administrator to update his policies to allow him to access the *`my-example-widget`* resource using the `account:`*`GetWidget`* action.

I am not authorized to perform `iam:PassRole`

If you receive an error that you're not authorized to perform the `iam:PassRole` action, your policies must be updated to allow you to pass a role to Account Management.

Some AWS services allow you to pass an existing role to that service instead of creating a new service role or service-linked role. To do this, you must have permissions to pass the role to the service.

The following example error occurs when an IAM user named `marymajor` tries to use the console to perform an action in Account Management. However, the action requires the service to have permissions that are granted by a service role. Mary does not have permissions to pass the role to the service.

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
 iam:PassRole
```

In this case, Mary's policies must be updated to allow her to perform the `iam:PassRole` action.

If you need help, contact your AWS administrator. Your administrator is the person who provided you with your sign-in credentials.

I want to allow people outside of my AWS account to access my account details

You can create a role that users in other accounts or people outside of your organization can use to access your resources. You can specify who is trusted to assume the role. For services that support resource-based policies or access control lists (ACLs), you can use those policies to grant people access to your resources.

To learn more, consult the following:

- To learn whether Account Management supports these features, see How AWS Account Management works with IAM.
- To learn how to provide access to your resources across AWS accounts that you own, see Providing access to an IAM user in another AWS account that you own in the *IAM User Guide*.
- To learn how to provide access to your resources to third-party AWS accounts, see Providing access to AWS accounts owned by third parties in the *IAM User Guide*.
- To learn how to provide access through identity federation, see Providing access to externally authenticated users (identity federation) in the *IAM User Guide*.
- To learn the difference between using roles and resource-based policies for cross-account access, see Cross account resource access in IAM in the *IAM User Guide*.

AWS managed policies for AWS Account Management

This information is most relevant for AWS accounts that you create when you use our advanced AWS experience or if you activated advanced features for your account. To learn about how access control works for our new AWS experience, see Compare access management.

AWS Account Management currently provides two AWS managed policies that are available for your use:

- AWS managed policy: AWSAccountManagementReadOnlyAccess
- AWS managed policy: AWSAccountManagementFullAccess
- Account Management updates to AWS managed policies

An AWS managed policy is a standalone policy that is created and administered by AWS. AWS managed policies are designed to provide permissions for many common use cases so that you can start assigning permissions to users, groups, and roles.

Keep in mind that AWS managed policies might not grant least-privilege permissions for your specific use cases because they're available for all AWS customers to use. We recommend that you reduce permissions further by defining customer managed policies that are specific to your use cases.

You cannot change the permissions defined in AWS managed policies. If AWS updates the permissions defined in an AWS managed policy, the update affects all principal identities (users, groups, and roles) that the policy is attached to. AWS is most likely to update an AWS managed policy when a new AWS service is launched or new API operations become available for existing services.

For more information, see AWS managed policies in the *IAM User Guide*.

AWS managed policy: AWSAccountManagementReadOnlyAccess

You can attach the `AWSAccountManagementReadOnlyAccess` policy to your IAM identities.

This policy provides read-only permissions to only view the following:

- The metadata about your AWS accounts
- The AWS Regions that are enabled or disabled for the AWS account (you can view status of Regions in your account only by using the AWS console)

It does this by granting permission to run any of the `Get*` or `List*` operations. It doesn't provide any ability to modify the account metadata or enable or disable AWS Regions for the account.

Permissions details

This policy includes the following permissions.

- `account` – Allows principals to retrieve the metadata information about AWS accounts. It also allows principals to list the AWS Regions that are enabled for the account in the AWS Management Console.

JSON

```
{
    "Version":"2012-10-17",
```

```
    "Statement": [
        {
            "Effect": "Allow",
            "Action": [
                "account:Get*",
                "account:List*"
            ],
            "Resource": "*"
        }
    ]
}
```

AWS managed policy: AWSAccountManagementFullAccess

You can attach the `AWSAccountManagementFullAccess` policy to your IAM identities.

This policy provides full administrative access to view or modify the following:

- The metadata about your AWS accounts
- The AWS Regions that are enabled or disabled for the AWS account (you can view status or enable or disable Regions for your account only by using the AWS console)

It does this by granting permission to run any `account` operations.

Permissions details

This policy includes the following permissions.

- `account` – Allows principals to view or modify the metadata information about AWS accounts. It also allows principals to list the AWS Regions that are enabled for the account and enable or disable them in the AWS Management Console.

JSON

```
{
    "Version":"2012-10-17",
    "Statement": [
        {
            "Effect": "Allow",
```

```
            "Action": "account:*",
            "Resource": "*"
        }
    ]
}
```

Account Management updates to AWS managed policies

View details about updates to AWS managed policies for Account Management since this service began tracking these changes. For automatic alerts about changes to this page, subscribe to the RSS feed on the Account Management Document history page.

Change	Description	Date
AWS Account Management launched with new AWS managed policies and started tracking changes	Account Management initially launched with the following AWS managed policies: • AWSAccountManagementReadOnlyAccess • AWSAccountManagementFullAccess	Sept 30, 2021

Compliance validation for AWS Account Management

Our new AWS experience cannot be used for any regulated workloads. The following information is only relevant to accounts created when you use Sign up for AWS (advanced).

Third-party auditors assess the security and compliance of AWS services that can run in your AWS account as part of multiple AWS compliance programs. These include SOC, PCI, FedRAMP, HIPAA, and others.

For a list of AWS services in scope of specific compliance programs, see AWS services in scope by compliance program. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading Reports in AWS Artifact in the *AWS Artifact User Guide*.

Your compliance responsibility when using services in your AWS account is determined by the sensitivity of your data, your company's compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying baseline environments on AWS that are security and compliance focused.
- Architecting for HIPAA Security and Compliance on Amazon Web Services – This whitepaper describes how companies can use AWS to create HIPAA-eligible applications.

  > **Note**
  >
  > Not all AWS services are HIPAA eligible. For more information, see the HIPAA Eligible Services Reference.

- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- Evaluating Resources with Rules in the *AWS Config Developer Guide* – The AWS Config service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub CSPM – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.
- AWS Audit Manager – This AWS service helps you continuously audit your AWS usage to simplify how you manage risk and compliance with regulations and industry standards.

Resilience in AWS Account Management

The following information is relevant for all AWS accounts. If you use our new AWS experience, additional Regional considerations apply. For more information, see AWS Regions for your projects.

The AWS global infrastructure is built around AWS Regions and Availability Zones. Regions provide multiple physically separated and isolated Availability Zones, which are connected through low-latency, high-throughput, and highly redundant networking. With Availability Zones, you can design and operate applications and databases that automatically fail over between zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

For more information about AWS Regions and Availability Zones, see AWS Global Infrastructure.

Infrastructure security in AWS Account Management

The following information is relevant for all AWS accounts.

As managed services, AWS services running in your AWS account are protected by the AWS global network security. For information about AWS security services and how AWS protects infrastructure, see AWS Cloud Security. To design your AWS environment using the best practices for infrastructure security, see Infrastructure Protection in *Security Pillar AWS Well-Architected Framework*.

You use AWS published API calls to access account settings through the network. Clients must support the following:

- Transport Layer Security (TLS). We require TLS 1.2 and recommend TLS 1.3.
- Cipher suites with perfect forward secrecy (PFS) such as DHE (Ephemeral Diffie-Hellman) or ECDHE (Elliptic Curve Ephemeral Diffie-Hellman). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the AWS Security Token Service (AWS STS) to generate temporary security credentials to sign requests.

Monitor your AWS account

This includes AWS accounts you create using our new AWS experience and accounts you create using our advanced AWS experience. To compare sign-up options, see Compare sign-up options.

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS Account Management and your other AWS solutions. AWS provides the following monitoring tools to watch Account Management, report when something is wrong, and take automatic actions when appropriate:

- *AWS CloudTrail* captures (logs) API calls and related events made by or on behalf of your AWS account and writes the log files to an Amazon Simple Storage Service (Amazon S3) bucket that you specify. This lets you identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred. For more information, see the AWS CloudTrail User Guide.
- *Amazon EventBridge* adds additional automation to your AWS services by responding automatically to system events, such as application availability issues or resource changes. Events from AWS services are delivered to EventBridge in near real time. You can write simple rules to indicate which events are of interest to you and which automated actions to take when an event matches a rule. For more information, see the Amazon EventBridge User Guide.

Logging AWS Account Management API calls using AWS CloudTrail

This includes AWS accounts you create using our new AWS experience and accounts you create using our advanced AWS experience. To compare sign-up options, see Compare sign-up options.

The AWS Account Management APIs are integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service that calls an Account Management operation. CloudTrail captures all Account Management API calls as events. The calls captured include all calls to the Account Management operations. If you create a trail, you can turn on continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for Account Management operations. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in **Event history**. Using the information collected by CloudTrail, you can determine the request that called an Account Management operation, the IP address used to make the request, who made the request and when, and additional details.

To learn more about CloudTrail, see the AWS CloudTrail User Guide.

Account Management information in CloudTrail

CloudTrail is turned on in your AWS account when you create the account. When activity occurs with an Account Management operation, CloudTrail records that activity in a CloudTrail event along with other AWS service events in **Event history**. You can view, search, and download recent events in your AWS account. For more information, see Viewing Events with CloudTrail Event History.

For an ongoing record of events in your AWS account, including events for Account Management operations, create a trail. A *trail* enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the AWS Management Console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. You can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for creating a trail
- CloudTrail supported services and integrations
- Configuring Amazon SNS notifications for CloudTrail
- Receiving CloudTrail log files from multiple Regions
- Receiving CloudTrail log files from multiple accounts

AWS CloudTrail logs all Account Management API operations found in the API Reference section of this guide. For example, calls to the `CreateAccount`, `DeleteAlternateContact`, and `PutAlternateContact` operations generate entries in the CloudTrail log files.

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

- Whether the request was made with root user or AWS Identity and Access Management (IAM) user credentials
- Whether the request was made with temporary security credentials for an IAM role or federated user
- Whether the request was made by another AWS service

For more information, see the CloudTrail userIdentity element.

Understanding the Account Management log entries

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested operation, the date and time of the operation, request parameters, and so on. CloudTrail log files aren't an ordered stack trace of the public API calls, so they don't appear in any specific order.

Example 1: The following example shows a CloudTrail log entry for a call to the `GetAlternateContact` operation to retrieve the current `OPERATIONS` alternate contact for an account. The values returned by the operation aren't included in the logged information.

Example 1


```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROA1234567890EXAMPLE:AccountAPITests",
    "arn":"arn:aws:sts::123456789012:assumed-role/ServiceTestRole/AccountAPITests",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROA1234567890EXAMPLE",
        "arn": "arn:aws:iam::123456789012:role/ServiceTestRole",
        "accountId": "123456789012",
        "userName": "ServiceTestRole"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-04-30T19:25:53Z"
      }
    }
  },
  "eventTime": "2021-04-30T19:26:15Z",
  "eventSource": "account.amazonaws.com",
  "eventName": "GetAlternateContact",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "10.24.34.250",
```

```
  "userAgent": "Mozilla/5.0",
  "requestParameters": {
    "alternateContactType": "SECURITY"
  },
  "responseElements": null,
  "requestID": "1a2b3c4d-5e6f-1234-abcd-111111111111",
  "eventID": "1a2b3c4d-5e6f-1234-abcd-222222222222",
  "readOnly": true,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "123456789012"
}
```

Example 2: The following example shows a CloudTrail log entry for a call to the `PutAlternateContact` operation to add a new `BILLING` alternate contact to an account.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROA1234567890EXAMPLE:AccountAPITests",
    "arn": "arn:aws:sts::123456789012:assumed-role/ServiceTestRole/AccountAPITests",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROA1234567890EXAMPLE",
        "arn": "arn:aws:iam::123456789012:role/ServiceTestRole",
        "accountId": "123456789012",
        "userName": "ServiceTestRole"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-04-30T18:33:00Z"
      }
    }
  },
  "eventTime": "2021-04-30T18:33:08Z",
  "eventSource": "account.amazonaws.com",
  "eventName": "PutAlternateContact",
```

```
  "awsRegion": "us-east-1",
  "sourceIPAddress": "10.24.34.250",
  "userAgent": "Mozilla/5.0",
  "requestParameters": {
    "name": "*Alejandro Rosalez*",
    "emailAddress": "alrosalez@example.com",
    "title": "CFO",
    "alternateContactType": "BILLING"
  },
  "responseElements": null,
  "requestID": "1a2b3c4d-5e6f-1234-abcd-333333333333",
  "eventID": "1a2b3c4d-5e6f-1234-abcd-444444444444",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "123456789012"
}
```

Example 3: The following example shows a CloudTrail log entry for a call to the `DeleteAlternateContact` operation to delete the current `OPERATIONS` alternate contact.

```
{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "AROA1234567890EXAMPLE:AccountAPITests",
    "arn":"arn:aws:sts::123456789012:assumed-role/ServiceTestRole/AccountAPITests",
    "accountId": "123456789012",
    "accessKeyId": "AKIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "AROA1234567890EXAMPLE",
        "arn": "arn:aws:iam::123456789012:role/ServiceTestRole",
        "accountId": "123456789012",
        "userName": "ServiceTestRole"
      },
      "webIdFederationData": {},
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2021-04-30T18:33:00Z"
      }
```

```
    }
  },
  "eventTime": "2021-04-30T18:33:16Z",
  "eventSource": "account.amazonaws.com",
  "eventName": "DeleteAlternateContact",
  "awsRegion": "us-east-1",
  "sourceIPAddress": "10.24.34.250",
  "userAgent": "Mozilla/5.0",
  "requestParameters": {
    "alternateContactType": "OPERATIONS"
  },
  "responseElements": null,
  "requestID": "1a2b3c4d-5e6f-1234-abcd-555555555555",
  "eventID": "1a2b3c4d-5e6f-1234-abcd-666666666666",
  "readOnly": false,
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "eventCategory": "Management",
  "recipientAccountId": "123456789012"
}
```


Monitoring Account Management events with EventBridge

This includes AWS accounts you create using our new AWS experience and accounts you create using our advanced AWS experience. To compare sign-up options, see Compare sign-up options.

Amazon EventBridge, formerly called CloudWatch Events, helps you monitor events that are specific to and initiate target actions that use other AWS services. Events from AWS services are delivered to EventBridge in near real time.

Using EventBridge, you can create *rules* that match incoming *events* and route them to *targets* for processing.

For more information, see Getting started with Amazon EventBridge in the *Amazon EventBridge User Guide*.

Account Management events

The following examples show events for Account Management. Events are produced on a best-effort basis.

Only events that are specific to enabling and disabling Regions and API calls via CloudTrail are currently available for Account Management.

Event types

- Event for enabling and disabling Regions

Event for enabling and disabling Regions

When you enable or disable a Region in an account, either from the Console or from the API, an asynchronous task is kicked off. The initial request will be logged as a CloudTrail event in the target account. In addition, an EventBridge event will be sent to the calling account when either the enable or disable process has started, and again once either process has completed.

The following example event shows how a request will be sent indicating that on `2020-09-30` the `ap-east-1` Region was `ENABLED` for account `123456789012`.

```
{
   "version":"0",
   "id":"11112222-3333-4444-5555-666677778888",
   "detail-type":"Region Opt-In Status Change",
   "source":"aws.account",
   "account":"123456789012",
   "time":"2020-09-30T06:51:08Z",
   "region":"us-east-1",
   "resources":[
      "arn:aws:account::123456789012:account"
   ],
   "detail":{
      "accountId":"123456789012",
      "regionName":"ap-east-1",
      "status":"ENABLED"
   }
}
```

There are four possible statuses which match the statuses returned by the `GetRegionOptStatus` and `ListRegions` APIs:

- `ENABLED` – The Region has been successfully enabled for the `accountId` indicated
- `ENABLING` – The Region is in the process of being enabled for the `accountId` indicated
- `DISABLED` – The Region has been successfully disabled for the `accountId` indicated

- `DISABLING` – The Region is in the process of being disabled for the `accountId` indicated

The following sample event pattern creates a rule that captures all Region events.

```
{
   "source":[
      "aws.account"
   ],
   "detail-type":[
      "Region Opt-In Status Change"
   ]
}
```

The following sample event pattern creates a rule that captures only `ENABLED` and `DISABLED` Region events.

```
{
   "source":[
      "aws.account"
   ],
   "detail-type":[
      "Region Opt-In Status Change"
   ],
   "detail":{
      "status":[
         "DISABLED",
         "ENABLED"
      ]
   }
}
```

Quotas for AWS Account Management

Your AWS account has default quotas, formerly referred to as limits, for each AWS service.

The following quota information is for projects you create if you use Sign up for AWS (new). To adjust a quota, create a support ticket. For more information, see Project support.

Resource	Quota	Adjustable
Number of projects that you own on the Free Plan	29	No
Number of projects you can own on the Paid Plan	299	No
Number of people you can share a project with	500	No
Number of projects invites you can accept	100	No

The following quota information is for AWS accounts you create if you sign up using Sign up (advanced). Unless otherwise noted, each quota is AWS Region-specific.

Resource	Quota
Maximum number of `StartPrimaryEmailUpdate` requests per target account	3 per 30 seconds
Number of alternate contacts in an AWS account	3 - one each for `BILLING`, `SECURITY`, and `OPERATIONS`
Number of concurrent region-opt requests per account	6
Number of concurrent region-opt requests per organization	50

Resource	Quota
Rate of `AcceptPrimaryEmailUpdate` requests per caller account	1 per second, burst to 1 per second
Rate of `DeleteAlternateContact` requests per account	1 per second, burst to 6 per second
Rate of `DisableRegion` requests per account	1 per second, burst to 1 per second
Rate of `EnableRegion` requests per account	1 per second, burst to 1 per second
Rate of `GetAccountInformation` requests per caller account	3 per second, burst to 3 per second
Rate of `GetAlternateContact` requests per account	10 per second, burst to 15 per second
Rate of `GetContactInformation` requests per account	10 per second, burst to 15 per second
Rate of `GetGovCloudAccountInformation` requests per account	3 per second, burst to 5 per second
Rate of `GetPrimaryEmail` requests per caller account	3 per second, burst to 3 per second
Rate of `GetRegionOptStatus` requests per account	5 per second, burst to 5 per second
Rate of `ListRegions` requests per account	5 per second, burst to 5 per second
Rate of `PutAccountName` requests per caller account	1 per second, burst to 1 per second
Rate of `PutAlternateContact` requests per account	5 per second, burst to 8 per second

Resource	Quota
Rate of `PutContactInformation` requests per account	5 per second, burst to 8 per second
Rate of `StartPrimaryEmailUpdate` requests per caller account	1 per second, burst to 1 per second

Document history for the Account Management User Guide

The following table describes the documentation releases for AWS Account Management.

Change	Description	Date
Updated account closure quota values	Corrected the account closure quota percentage from 10% to 20% and the minimum from 10 to 250 in the troubleshooting topic.	June 23, 2026
New account name APIs	Support for new `GetAccountInformation`, and `PutAccountName` APIs to view or modify an account name.	April 22, 2025
End of support for editing security challenge questions	Removed the *Edit your security challenge questions* topic from the guide since support has ended.	January 6, 2025
New primary email APIs	Support for new `GetPrimaryEmail`, `StartPrimaryEmailUpdate`, and `AcceptPrimaryEmailUpdate` APIs to centrally update the root user email address for any member account in AWS Organizations. For more information, see Updating the root user email address for a member	June 6, 2024

	account in the *AWS Organizations User Guide*.	
Rewrite of the close account topic	Completely overhauled the entire close account topic including adding steps for how to close member and management accounts.	February 1, 2024
End of support for adding new security challenge questions	Added new content noting that the option to add new challenge questions was removed from the **Accounts** page.	January 5, 2024
End of support for the `aws-portal` namespace	AWS Identity and Access Management (IAM) actions that were previously used to manage your account (for example, `aws-portal:ModifyAccount` and `aws-portal:ViewAccount`) have reached the end of standard support.	January 1, 2024
Rewrite of the Regions topic	Completely overhauled the entire Regions topic including adding expand and collapse controls.	October 8, 2023
Relocated root user topics to the IAM User Guide	Consolidated discussion about root users into one topic, added cross reference links to root user topics that were moved to the IAM User Guide.	September 18, 2023

New section added to the primary account contact topic	Added new *Phone number and email address requirements* section.	September 12, 2023
New contact information APIs	Support for new `GetContactInformation` and `PutContactInformation` APIs.	July 22, 2022
AWS Account Management now supports updating alternate contacts via the AWS Organizations console.	You can now update your organization's alternate contacts via AWS Organizations console using Account API permissions provided by updated AWS Organizations managed policies.	February 8, 2022
Initial release	Initial release of the AWS Account Management Reference Guide	September 30, 2021