AWS Whitepaper

Organizing Your AWS Environment Using Multiple Accounts

Organizing Your AWS Environment Using Multiple Accounts: AWS Whitepaper

Table of Contents

Organizing Your AWS Environment Using Multiple Accounts

Publication date: **April 30, 2025** (*Document history*)

Businesses that are starting to adopt Amazon Web Services (AWS), expanding their footprint on AWS, or planning to enhance an established AWS environment need to ensure they have a foundation on AWS for their cloud environment. One important aspect of their foundation is organizing their AWS environment following a multi-account strategy.

Using multiple AWS accounts to help isolate and manage your business applications and data can help you optimize across most of the AWS Well-Architected Framework pillars, including operational excellence, security, reliability, and cost optimization. This paper provides best practices and current recommendations for organizing your overall AWS environment. The extent to which you use these best practices depends on your stage of the cloud adoption journey and specific business needs.

Are you Well-Architected?

The AWS Well-Architected Framework helps you understand the pros and cons of the decisions you make when building systems in the cloud. The six pillars of the Framework allow you to learn architectural best practices for designing and operating reliable, secure, efficient, cost-effective, and sustainable systems. Using the AWS Well-Architected Tool, available at no charge in the AWS Management Console, you can review your workloads against these best practices by answering a set of questions for each pillar.

For more expert guidance and best practices for your cloud architecture—reference architecture deployments, diagrams, and whitepapers—refer to the AWS Architecture Center.

Introduction

Using multiple AWS accounts to help isolate and manage your business applications and data can help you optimize across most of the AWS Well-Architected Framework pillars including operational excellence, security, reliability, and cost optimization.

Multi-account strategy best practices and recommendations

Businesses can benefit from considering the latest guidance for organizing their AWS environments. A multi-account strategy is key to succeed when customers are starting to adopt AWS, expanding their footprint on AWS, or planning to enhance an established AWS environment.

Customers might have multiple teams with different security and compliance controls that need to be isolated from one another. Some might have different business processes entirely or be part of different business lines that need clarity around costs incurred.

Customers need explicit security boundaries, a mechanism to have direct control and visibility of their limits and any throttling, and a billing separation to directly map costs to underlying projects. The isolation designed into an AWS account can help you meet these needs.

Using multiple AWS accounts to help isolate and manage your business applications and data can help you optimize across most of the AWS Well-Architected Framework pillars including operational excellence, security, reliability, and cost optimization.

AWS accounts

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.

The number of accounts that best meets your needs can range from a few to hundreds or even thousands. Management of many accounts requires use of automation to help minimize your Multi-account strategy best practices and recommendations 2 operational complexity and ensure efficient alignment with your security, governance, and operational requirements. AWS does not charge per account. Rather, you incur charges based on resources used, regardless of account quantity.

Rather than using a single account, we recommend that you use several accounts to separate your workloads. A workload identifies a set of components that deliver business value together. A workload is usually the level of detail that business and technology leaders communicate about. This approach is designed to make it easier for you to meet your requirements, even in the early

project stage of adoption. Based on the success of those first few workloads, you might want to gain further business benefits by expanding your adoption of AWS. This motivation often leads to the foundation stage of adoption. In this stage, you invest in evolving your cloud foundational capabilities before greatly expanding adoption.

A workload identifies a set of components that deliver business value together. A workload is usually the level of detail that business and technology leaders communicate about. Some examples of workloads are:

- Marketing websites
- Ecommerce websites
- Mobile app backends
- Analytic platforms

Workloads vary in levels of architectural complexity, from static websites to complex microservices, each with potentially different requirements on cost or billing identification.

Stages of adoption

When initially adopting AWS, it's recommended to use a multi-account strategy rather than relying on a single account. This approach is designed to provide flexibility and scalability, even in the early stages of your cloud journey. By separating resources into distinct accounts from the start, you can more easily manage security policies, access controls, and cost optimization as your initial workloads are migrated to the cloud. This lays the groundwork for a secure and well-governed AWS environment.

As your initial cloud projects demonstrate success, the motivation often arises to expand AWS adoption further across the organization. This leads to the foundation stage of the cloud adoption lifecycle. During this phase, you invest in evolving your core cloud capabilities, such as establishing center of excellence teams, automating provisioning, and implementing cost management best practices. Building this strong cloud foundation positions you for widespread enterprise-wide adoption down the line, allowing you to capitalize on the full breadth of benefits that the AWS platform can provide.

Best practices

The best practices described in this paper are designed to help you more easily achieve your security, governance, and operational requirements through multiple accounts. The best practices

were assembled based on the experiences of thousands of customers who have progressed through their cloud adoption journeys.

The best practices can help you quickly establish the initial cloud foundation of your AWS environment, and adjust and expand your AWS environment as you gain experience both with the AWS services and how you work with the AWS Cloud.

While organizations often have similar cloud adoption needs, each has unique requirements. Therefore, these AWS multi-account best practices are intended as guidance, not a one-size-fits-all solution. Your specific AWS environment design may differ from the examples provided, but following these practices will help you make informed decisions as you plan your cloud strategy.

Relation to AWS Well-Architected

AWS Well-Architected helps cloud architects build secure, high-performing, resilient, and efficient infrastructure for their applications and workloads. Based on six pillars—operational excellence, security, reliability, performance efficiency, cost optimization, and sustainability—AWS Well-Architected provides a consistent approach for customers and partners to evaluate architectures and implement designs that can scale over time.

The best practices for organizing your AWS environment addressed in this guide augment and support the best practices represented in the Well-Architected pillars.

Intended audience

These best practices are intended for cloud architects and technical leads who are responsible for the overall security and architecture of an AWS environment. Whether you are new to AWS or you have already been using AWS for years, your team will benefit from reviewing these best practices and comparing them to your requirements and current AWS environment.

These best practices are intended to apply to organizations largely independent of their industry, size, expected scale of adopting AWS, and workload portfolio. Depending on your needs, not all of the best practices might apply to your situation.

If you're just starting to experiment and learn about AWS by using a single AWS account, you don't need to consider these best practices until you begin planning for your first few production workloads.

Benefits of using multiple AWS accounts

As you adopt AWS, we recommend that you determine how your business, governance, security, and operational requirements can be met in AWS. Use of multiple AWS accounts plays an important role in how you meet those requirements.

The use of multiple accounts enables you to realize the benefits in the following sections.

Topics

Group workloads based on business purpose and ownership

You can 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.

Different business units or product teams might have different processes or security boundary requirements. Depending on your overall business model, you might choose to isolate distinct business units or subsidiaries in different accounts or Organizational Units. Isolation of business units can help them operate with greater decentralized control, but still provides the ability for you to provide overarching controls. This approach might also ease divestment of those units over time.

Controls are governance rules for security, operations, and compliance that you can define and apply to align with your overall requirements.

If you acquire a business that is already operating in AWS, you can move the associated accounts intact into your existing organization. This movement of accounts can be an initial step toward integrating acquired services into your standard account structure.

Apply distinct security controls by environment

Workloads often have distinct security postures that require separate controls and mechanisms to support them. For example, it's common to apply different security and operational controls for the non-production and production environments of a given workload. By using separate accounts for the non-production and production environments, by default, the resources and data that make up a workload environment are separated from other environments and workloads.

Constrain access to sensitive data

A key benefit of a multi-account architecture is the ability to create distinct data perimeters. By limiting sensitive data stores to dedicated accounts, you can more tightly control access and minimize the number of people and processes that can interact with that data. This simplified access model aligns with the principle of least privilege, as you can restrict permissions at the coarse-grained account level rather than needing to manage fine-grained access within a single broad account.

Establishing these data perimeters is particularly valuable for highly confidential or regulated information. By segmenting sensitive data into its own secured account, you can drastically reduce the exposure surface and attack vectors. This account-level isolation helps contain potential data breaches and ensures that even in the event of a security incident, the blast radius remains limited to that specific data domain rather than impacting your entire cloud environment.

For example, designating a set of accounts to house publicly accessible (Amazon S3) buckets enables you to implement policies for all of your other accounts to prohibit making Amazon S3 buckets publicly available.

Promote innovation and agility

In the early stages of a workload's lifecycle, you can help promote innovation by providing your builders with separate accounts in support of experimentation, development, and early testing.

These environments often provide greater freedom than more tightly controlled production-like test and production environments by enabling broader access to AWS services while using controls to help prohibit access to and use of sensitive and internal data.

- **Sandbox accounts** are typically disconnected from your enterprise services and do not provide access to your internal data, but offer the greatest freedom for experimentation.

- **Development accounts** typically provide limited access to your enterprise services and development data, but can more readily support day-to-day experimentation with your enterprise approved AWS services, formal development, and early testing work.

In both cases, we recommend security controls and cost budgets so that you limit risks and proactively manage costs.

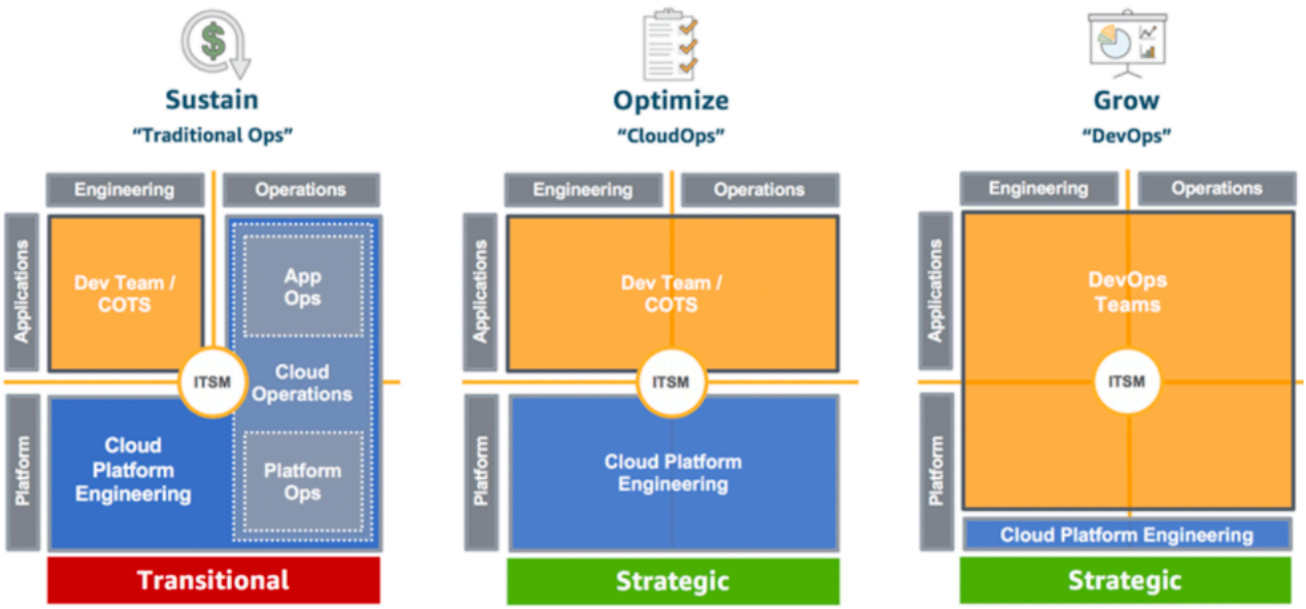
In support of later stages of the workload lifecycle, you can use distinct test and production accounts for workloads or groups of related workloads. Having an environment for each set of workloads can enable owning teams to move faster by reducing dependencies on other teams and workloads and minimizing the impact of changes.

Limit scope of impact from adverse events

The inherent isolation provided by an AWS account can be leveraged to limit the scope of impact from issues within your cloud environment. By provisioning resources within dedicated accounts, you establish clear security, access, and billing boundaries that prevent problems from spreading between different parts of your infrastructure. This account-level isolation is a key benefit of a multi-account strategy, as it helps ensure that if an application-level problem, misconfiguration, or malicious activity occurs within one account, the impacts can be contained and prevented from impacting workloads in other accounts. Maintaining this resource independence across multiple accounts is a powerful way to manage risk and help ensure failures remain localized, providing a safeguard against wider disruptions that could occur in a single monolithic account environment.

Support multiple IT operating models

Organizations often have multiple IT operating models or ways in which they divide responsibilities among parts of the organization to deliver their application workloads and platform capabilities. The following figure shows three example operating models:



Example operating models

- In the *Traditional Ops* model, teams who own custom and commercial off-the-shelf (COTS) applications are responsible for engineering their applications, but not for their production operations. A cloud platform engineering team is responsible for engineering the underlying platform capabilities. A separate cloud operations team is responsible for the operations of both applications and platform.
- In the *CloudOps model*, application teams are also responsible for production operations of their applications. In this model, a common cloud platform engineering team is responsible for both engineering and operations of the underlying platform capabilities.
- In the *DevOps model*, the application teams take on the additional responsibilities of engineering and operating platform capabilities that are specific to their applications. A cloud platform engineering team is responsible for engineering and operations of shared platform capabilities that are used by multiple applications.

As a practice, IT Service Management (ITSM) is a common element across all of the models. Your overall goals and requirements of ITSM might not change across these models, but the responsible individuals and solutions for meeting those goals and requirements can vary depending on the model.

Given the implications of centralized operations versus more distributed operational responsibilities, you will likely benefit from establishing separate groups of accounts in support of different operating models. Use of separate accounts enables you to apply distinct governance and operational controls that are appropriate for each of your operating models. To learn more about operating models and their implications on your cloud adoption, refer to the AWS Well-Architected Operational Excellence Pillar Operating Model.

Manage 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 report, control, forecast, and budget your cloud expenditures.

In addition to cost reporting at the account level, AWS has built-in support to consolidate and report costs across your entire set of accounts. When you require fine-grained cost allocation, you can apply cost allocation tags to individual resources in each of your accounts. For more information about cost optimization, see the AWS Well-Architected Cost Optimization Pillar's Expenditure and Usage Awareness best practices.

Distribute AWS Service Quotas and API request rate limits

AWS Service Quotas, also known as limits, are the maximum number of service resources or operations that apply to an account. For example, the number of Amazon S3 buckets that you can create for each account.

You can use Service Quotas to help protect you from unexpected excessive provisioning of AWS resources and malicious actions that could dramatically impact your AWS costs.

AWS services can also throttle or limit the rate of requests made to their API operations. Because Service Quotas and request rate limits are allocated for each account, use of separate accounts for workloads can help distribute the potential impact of the quotas and limits.

To learn more about managing service quotas, refer to the AWS Well-Architected Reliability Pillar: Manage Service Quotas and Constraints.

Core concepts

This section covers the following core concepts for defining your multi-account strategy on AWS:

- AWS Organizations
- Benefits of using organizational units (OUs)
- Multiple organizations

AWS Organizations

AWS Organizations helps you centrally govern your environment as you grow and scale your workloads on AWS. Whether you are a growing startup or a large enterprise, Organizations helps you to centrally provision accounts and resources; secure and audit their environment for compliance; share resources; control access to accounts, Regions, and services; as well as optimize costs and simplify billing. Additionally, Organizations supports aggregation of health events, consolidated data on use of access permissions, and centralized management of backups and tagging for multi-account environments.

This section includes best practices for organizing your AWS accounts, including grouping your accounts into organizational units (OUs) so that you can more effectively secure and manage your overall AWS environment.

What is an Organization?

An organization is an entity that you create to consolidate a collection of accounts so that you can administer them as a single unit. Within each organization, you can organize the accounts in a hierarchical, tree-like structure with a root at the top and organizational units (OUs) nested under the root. Each account can be placed directly in the root, or placed in one of the OUs in the hierarchy.

Each organization consists of:

- A management account
- Zero or more member accounts
- Zero or more organizational units (OUs)

- Zero or more policies

Organizations management account

The management account creates the AWS organization's resources, OUs, and policies, to manage the organization's member accounts. Access to the management account must be strictly controlled by a small set of highly-trusted individuals from the organization, following the Principles of Least Privilege based on the activities they need to perform. This account **should not** be used for business workloads and should not contain business resources.

Additionally, the organization management account is where automation tooling is installed to automate consistent deployment of controls or other standardized infrastructure constructs across accounts in an organization. A trust relationship, which is used by the automation tooling, exists between child AWS accounts in the organization and the organization management account.

This relationship is established by default when new AWS accounts are created in the organization, and it enables management account users and roles to assume this cross-account AWS Identity and Access Management (IAM) role in child accounts.

Considerations for setting up the management account:

Most customers start with one AWS account, where they build some Proof of Concepts (PoCs) before deploying their workloads on AWS. In this situation, we recommend creating a new AWS account to be your management account, and inviting your existing account into your new AWS organization. This allows you to keep any PoCs or workloads that you might already have in that account intact.

When you set up the management account, we recommend using an email address that belongs to a shared mailbox, to avoid losing access to this account if only one individual has access to this email address, and for example, they leave your organization or lose access to the account.

Organizations member accounts

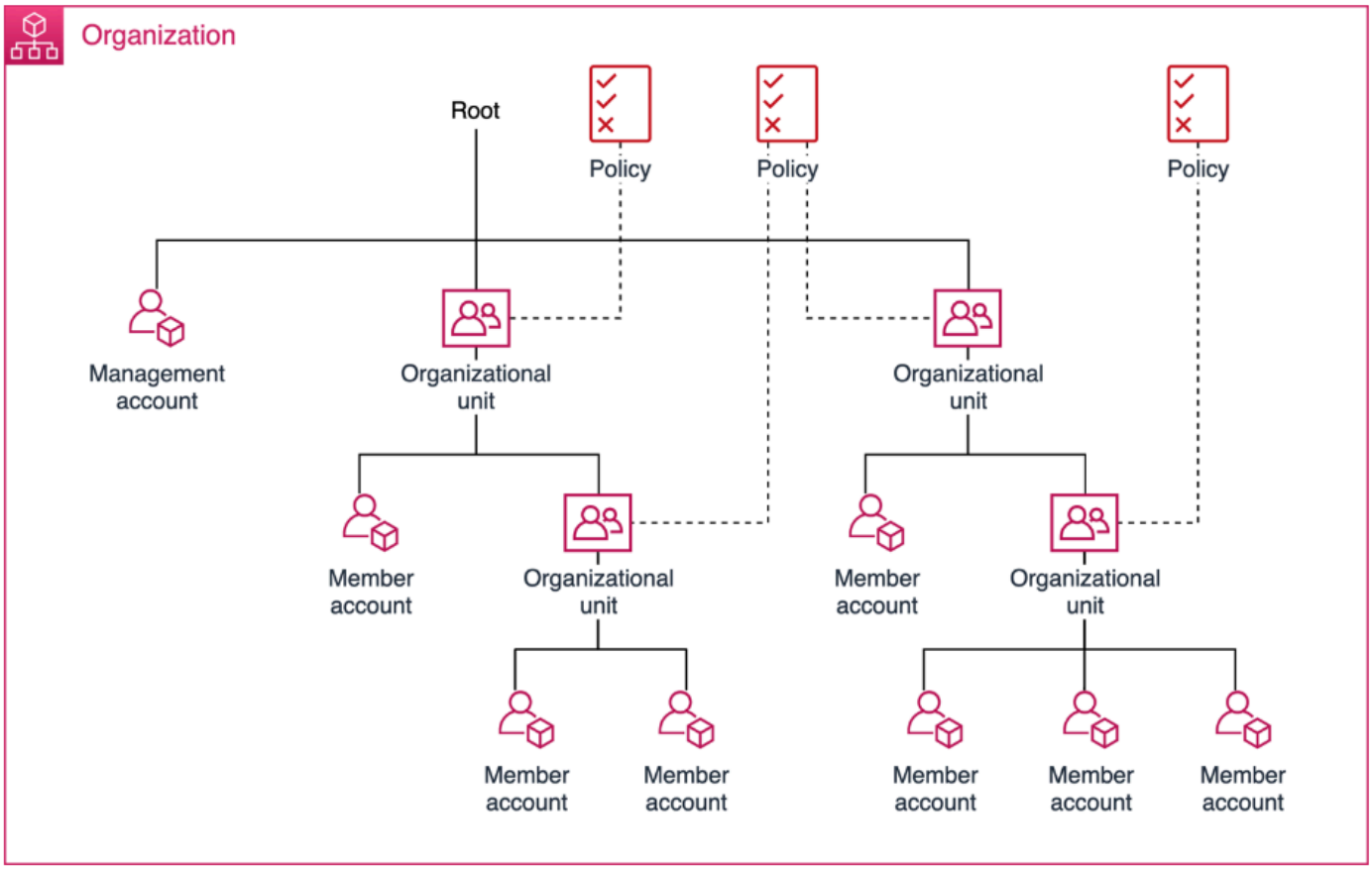
AWS Organizations member accounts belong to the organization and reside in the overall organization's structure. All billing for member accounts is consolidated to the management account of the organization. Most of your workloads will reside in member accounts, except for some centrally managed processes that must reside in either the management account or in accounts assigned as designated administrators for specific AWS services.

Organizational units (OUs)

An organizational unit (OU) provides a means to group accounts within a root. An OU can also contain other OUs. When you attach a policy to one of the nodes in the hierarchy, it flows down and affects all the branches (OUs) and leaves (accounts) beneath it. An OU can have exactly one parent, and each account can be a member of exactly one OU.

OUs are not meant to mirror your own organization's reporting structure. Instead, OUs are intended to group accounts that have common overarching security policies and operational needs. The primary question to ask yourself is: How likely will the group need a set of similar policies?

The following diagram shows a basic organization that consists of seven accounts that are organized into four OUs under the root. The organization also has a few policies that are applied to OUs.



Example of a basic organization

Benefits of using organizational units (OUs)

The following benefits of using OUs helped shape Recommended OUs and accounts.

Group similar accounts based on function

When you have multiple accounts that perform either similar or related functions, you can benefit from grouping these accounts into distinct top-level OUs. Prudent use of top-level OUs can help your teams better understand the overall structure of your AWS accounts.

For example, these best practices recommend a set of top-level OUs to help you organize different sets of related accounts. At a minimum, the top-level OUs are used to distinguish between overall functions of accounts.

Apply common policies

OUs provide a way for you to organize your accounts so that it's easier to apply common overarching policies to accounts that have similar needs. Policies in AWS Organizations enable you to apply additional types of management to the accounts in your organization.

By attaching policies to OUs rather than to individual accounts, you can simplify management of policies across groups of similar accounts. As the number of accounts in your environment grows, simplifying policy management by attaching policies to OUs becomes more important.

AWS Organizations supports use of authorization and management policies. For a complete list of policy types, refer to Managing AWS Organizations policies.

Authorization policies

- **Service control policies (SCPs):** SCPs are a type of organization policy that you can use to manage permissions in your organization. SCPs offer central control over the maximum available permissions for all accounts in your organization. SCPs are a means of implementing controls in your AWS organization. Your use of SCPs can help ensure that your accounts stay within your access control guidelines. For example, you can use SCPs to constrain the set of AWS services and actions allowed on resources.
  - Use SCPs to restrict your AWS Organizations' IAM principals' access to services and resources, or the conditions under which IAM principals can make requests. Refer to SCP examples and guidance on when to use SCPs.

- **Resource control policies (RCPs):** RCPs are another type of authorization policy that you can use to manage permissions in your organization. RCPs offer central control over the maximum available permissions for resources in your organization. RCPs help you to ensure resources in your accounts stay within your organization's access control guidelines.
  - Use RCPs to restrict who can access your resources, and enforce requirements on how your resources can be accessed. For example, only trusted external accounts can access specific Amazon S3 buckets hosted in your organization. Refer to RCP examples and guidance on when to use RCPs.

Although you can apply SCPs and RCPs to the root of your organization, you typically associate these policies with underlying OUs. For example, based on the nature of the workloads deployed in accounts within an OU, you might choose to restrict the set of AWS services and AWS Regions that are allowed to be used by accounts in the OU.

The effective permissions are the logical intersection between what is allowed by the RCPs and SCPs, and what is allowed by the identity-based and resource-based policies.

> **Note**
>
> Authorization policies (SCPs and RCPs) don't affect users, roles, or resources in the management account. They affect only the member accounts in your organization. This also means that SCPs and RCPs apply to member accounts that are designated as delegated administrators for policy management.

Management policies

You can also apply the following policies to your organization:

- Tag policies: Tag policies can help you monitor and ensure compliance with your cloud resource tagging standards.
- AI services opt-out policies: You can use artificial intelligence (AI) services opt-out policies to control data collection for AWS AI services for all of your organization's accounts.
- Backup policies: Backup policies can help you centrally manage and apply backup plans to the AWS resources across your organization's accounts.

- Chatbot policies: Chatbot policies can help you to control access to your organization's accounts from chat channels. You can use Chatbot policies to determine which permissions models, chat platforms and chat workspaces can be used to access the accounts.
- Declarative policies for EC2: Declarative policies allow you to centrally declare and enforce your desired configuration for a given AWS service at scale across an organization. Once attached, the configuration is always maintained when the service adds new features or APIs.
  - Use declarative policies to prevent non-compliant actions. For example, you can block internet access to Amazon VPC resources across your organization.

Share common resources

OUs provide a means for you to organize your accounts so that it's easier for you to share centrally managed resources across similar accounts.

AWS services have been introducing support for sharing their resources through AWS Resource Access Manager (AWS RAM) and AWS Organizations. For example, with AWS RAM, you can use OUs as the basis for sharing centrally managed network resources such as Amazon Virtual Private Cloud (Amazon VPC) subnets.

Provision and manage common resources

Sometimes you need to deploy common, centrally managed resource configurations to groups of related accounts. In cases where resource sharing doesn't apply, you can use a variety of AWS services and third-party tools that work with OUs to automatically roll out and update your own custom resources.

For example, you can use OUs as a basis for targeting automation to deploy and update your own sets of IAM roles and customer managed IAM policies that help establish common baseline and/or workload-specific security controls to groups of related accounts.

Multiple organizations

Most customers are best served using a single production organization and a test organization. This allows you to ensure consistency across accounts in your environment because centrally-applied policies or service-level configurations are programmatically applied across accounts within your organization. Separating your workload accounts across organizations requires additional overhead or customization to ensure central standards are applied within each organization.

There are certain exceptions, outlined in this section, where you might need to work across multiple organizations.

Test changes to your overall AWS environment

It is recommended to develop infrastructure as code that interacts with APIs and other mechanisms fundamental to the management of your AWS Organization. In these cases, to determine whether your changes break something without having to make the changes in your production organization, we recommend that you test your changes in an organization different from the one running your production workloads.

For example, you might need to either modify the automation that creates new accounts to change the configuration baseline of accounts it creates or change the configuration of a workflow management system you're using to modify SCPs. In addition, you might want to test the delegated administration capabilities of various AWS services prior to applying them in your production organization.

In these circumstances, we recommend that you establish an additional AWS Organization for testing that closely resembles your production organization. You can perform testing of changes to how you manage your organization in your test organization before applying those changes to be applied to your production organization.

Support acquisitions and divestments

You might acquire an entity that has already established an organization. If you decide to merge the acquired entity's AWS environment with your AWS environment, you can move member accounts from the acquired organization to your mainstream organization. In this case, you can later decommission the acquired entity's organization.

If you plan to potentially divest a portion of your portfolio, you can manage the workloads and supporting AWS accounts for that portion of your portfolio in a separate organization to support simpler divesture and isolated billing.

Support large AWS environments

You can request a quota increase if your AWS Organization reaches its maximum account limit. Support can help expand your organization's capacity beyond the default number of accounts. For more details about the maximum number of accounts supported in an organization, refer to Quotas for AWS Organizations.

Align with your billing requirements

An organization gathers billing information from all member accounts into a single consolidated AWS bill. If you have use cases where different sets of accounts require distinct bills or payments, then multiple organizations might be required. Ensure you have reviewed AWS Billing and Cost Management Conductor and Billing and Management Credit sharing options for your AWS accounts to understand the configurations that are supported within a single AWS Organization.

Different classification levels for government applications

Some customers in government, critical national infrastructure providers, and defense industries need to handle data with defined classification levels. These customers require high-assurance mechanisms to keep data (and, in most cases, metadata) associated with at least some of those markings, separate from each other. Assets within a single account should all handle data at the same protective marking.

As noted in this section, an organization itself contains collective data from multiple accounts. Data such as account names, billing data, organizational unit names, and activity logs can be accessed centrally for those with appropriate permissions, such as a cloud administrator or audit team. This means that commingling of billing and logging data from accounts that are processing data might not meet a customer's requirements for separation by classification levels.

An organization's configuration could be modified to have customized separation for protective marking with proper tags, logging customization (based on protective markings), and defined permissions for administrative users. However, this might be more easily achieved with separate organizations.

Design principles for your multi-account strategy

The following design principles helped develop the best practices described in this paper. You can also use these principles to help guide your initial account design and evolve it over time.

These design principles complement the Benefits of using multiple accounts and Benefits of using OUs.

Topics

Organize based on security and operational needs

We recommend that you organize accounts using OUs based on function, compliance requirements, or a common set of controls rather than mirroring your organization's reporting structure.

Apply security controls to OUs rather than accounts

Where feasible, we recommend that you apply security controls (for example, SCPs) to OUs instead of accounts so that you can more efficiently manage the distribution of controls across accounts that have the same or similar requirements.

For more information about managing security controls, refer to Permissions management in the AWS Well-Architected Security Pillar.

Avoid deep OU hierarchies

Overly complicated structures can be difficult to understand and maintain. Although AWS Organizations supports a depth of five levels of OUs, the recommended structure strives to use OUs only when there is sufficient benefit.

When you consider the addition of new OU levels, you should review the Benefits of using OUs and these principles to decide whether the additional complexity adds sufficient value.

Start small and expand as needed

We recommend that you start with a subset of the Recommended OUs and accounts and expand the structure of your AWS accounts when your needs call for the creation of new OUs.

You shouldn't need to invest a lot of time at the beginning of your adoption journey designing what you expect your AWS account structure will look like in several years.

Avoid deploying workloads to the organization's management account

Since privileged operations can be performed within an organization's management account and SCPs do not apply to the management account, we recommend that you limit access to an organization's management account. You should also limit the cloud resources and data contained in the management account to only those that must be managed in the management account.

Many AWS services that integrate with Organizations enable you to reduce the usage of the management account. These services enable you to register one or more-member accounts as administrators that can manage all of the organization's accounts used in the service. These accounts are called delegated administrators for that specific service. By registering a member account as a delegated administrator for an AWS service you enable that account to have some administrative permissions for that service, reducing the number of users that require management account access.

Separate production from non-production workloads

We recommend that you separate production workloads from non-production workloads. For overall recommendations on designing this separation, refer to Organizing workload-oriented OUs.

Assign a single or small set of related workloads to each production account

In support of your production workloads, we recommend that you either assign a single workload to each production account or assign a small set of closely related workloads to each production account.

Consider separating workloads that have different owners into their own production accounts to simplify access management, streamline change approval processes, and limit the scope of impact for misconfigurations.

Use federated access to help simplify managing human access to accounts

We recommend that you use AWS identity federation capabilities by using AWS IAM Identity Center. These capabilities enable you to use a common identity provider and your existing processes for controlling human user access to your AWS accounts.

By using federated access and a common identity provider, you avoid the need to manage individual users in each account. Instead, your human users can use their existing credentials to access authorized accounts. You also gain the benefit of keeping personally identifiable information (PII) out of IAM.

With federated access, your human users use temporary credentials instead of long-term access keys for programmatic access to their AWS environments.

Use of federated access avoids the creation and management of users in your AWS accounts for humans. Instead, use of users can be limited to those exceptional cases such as third-party applications that do not support the use of roles.

For more information about managing identities, refer to Identity Management in the AWS Well-Architected Security Pillar and Identity federation in AWS.

Use automation to support agility and scale

It is important to design and manage your accounts so that you can rapidly respond to business needs without the need for a corresponding linear increase in headcount. When you consider moving beyond managing just a few accounts, you must consider the work to establish processes and automation that will enable you to do so in an efficient manner.

For example, if you implement an account design in which new business initiatives call for the creation of new accounts, then you will benefit from having automation in place so that you can rapidly and reliably provision environments based on your standard configurations. Automation can also help you monitor compliance and apply updates to your baseline configurations over time.

Use multi-factor authentication

Multi-factor authentication (MFA) should be used by your root and all AWS users in your accounts regardless of privilege level or access mechanism. You can follow our current recommendation for MFA best practices for your AWS accounts (management account or member accounts) to set up MFA across your AWS environment.

Multiple AWS Regions

If you plan to use multiple AWS Regions, keep the following considerations in mind as you design your overall AWS environment.

Topics

Geographic scopes of data protection

If you use different AWS Regions that are in the same geographic scope defined by the data protection requirements applicable to your workloads, you can use the same IAM IdP or IdPs to federate to all accounts in live, disaster recovery, or load balanced live environments. You can replicate databases between environments using appropriate mechanisms, such as Amazon DynamoDB global tables or Amazon RDS read replicas. In such circumstances, it is also possible for

you to distribute core elements of your foundational AWS environment such that the log archive bucket is in one Region and assets in other accounts in other Regions log cross-Region to it.

You should carefully consider whether the data protection requirements applicable to your workload differ across countries, or are subject to data sovereignty requirements or export control.

This might impact your ability to make cross-Region data transfers. (Note that cross-Region data transfers incur networking costs.)

Performance considerations

There are also performance considerations to keep in mind for certain workloads. Some services are by their nature per-Region, which makes it more sensible for you to deploy such workloads with all assets in the same Region. For example, AWS KMS keys cannot be exported from a Region, and use of a KMS key in another Region is likely going to add latency to an application. We therefore recommend using AWS KMS in the same Region, unless specific governance policies, regulatory or corporate, mandate otherwise.

Close collaboration between your security and architecture teams and your workload owning teams is important to properly using KMS. Your design of how Amazon S3 objects, EBS volumes, and other data are encrypted and potentially replicated across Regions should factor in low latency when required.

Where cross-account replication of these assets is required, Amazon S3 Cross-Region Replication (CRR) enables on-the-fly re-encryption of an object with an AWS KMS key in the destination Region. Multi-Region duplication of AWS KMS keys for the decryption of cross-Region copied EBS volumes can be achieved using the techniques covered in Busy Engineer's Document Bucket.

Log management

When logs are generated, we recommend that you implement secondary controls to filter them before they are passed outside a compliance scope boundary associated with an account, or are passed cross-Region. If your logs contain sensitive data, this approach helps ensure that such sensitive data cannot escape your defined compliance scope boundary using AWS logging capabilities.

Although AWS CloudTrail has built-in cross-account logging capability and AWS Config can aggregate configuration and compliance data across accounts and Regions, it might be more appropriate for you to aggregate logs in an account. You can use AWS Lambda functions or similar

to filter the logs before sending them to another Region for aggregation into a multi-Region logging archive.

Break glass access

The organization management account is used to provide break glass access to AWS accounts within the organization. Break glass (which draws its name from breaking the glass to pull a fire alarm) refers to a quick means for a person who does not have access privileges to certain AWS accounts to gain access in exceptional circumstances by using an approved process.

The use cases for break glass access include:

- Failure of the organization's IdP.
- A security incident involving the organizations' IdP(s).
- A failure involving IAM Identity Center.
- A disaster involving the loss of an organization's entire cloud or IdP teams. It is important that access to these roles is monitored, and alarms and alerts are triggered when the roles are used to access the environment.

In the case of an incident requiring remediation, we recommend that a user with access to an administrative federated role within the AWS account perform the required remediation. In cases where this user is unavailable to carry out a time sensitive action, we recommend that a highly-restricted group or set of groups be preconfigured within your IdP, each providing appropriate federated access into the appropriate set of AWS accounts. A user can either be added into one of these groups using a high-priority and temporary change request, or a select group of privileged and trusted users can be prepopulated into these groups.

Security teams investigating an incident would use this mechanism to access a read-only role in an impacted account, or use the read-only access mechanism provided through the security tooling account. In summary, common high-priority irregular access scenarios need to be incorporated into standard federated access processes and procedures.

> **Note**
>
> AWS Organizations Service Control Policies do not apply to the organization management account, and administrator access to this account would grant privileged status to the entire organization, given the trust relationship to the management account. Therefore,

access to break glass IAM users must be tightly controlled, but accessible through a predefined and strict process. This process often involves one trusted individual having access to the password, and a different trusted individual having access to the hardware multi-factor authentication (MFA) key, meaning it typically requires two people to access any one set of break glass credentials.

Human access to AWS accounts within the organization should be provided using federated access. Although the use and creation of AWS IAM users is highly discouraged, break glass users are an exception.

To ensure human break-glass access to your environment, we recommend that you create the following in your AWS organization:

- At least two IAM users with IAM login credentials to prevent lockdown in case one of them is not available, and additional users depending on your operating model. Do not create unnecessary IAM privileged users in your management account. These users will assume roles in the member accounts in your organization through trust policies.
- A break glass role that is deployed to all the accounts in the organization, and that can only be assumed by the break glass users from the management account. These roles are needed to allow access from the management account to apply and update controls, to troubleshoot and resolve issues with the automation tooling from the security tooling account, or to remediate security and operational issues in one of the member accounts in the AWS organization. When setting up these roles in your organization, you need to ensure they can be used in emergency situations, bypassing established controls under the situations described earlier in the paragraph, such as service control policies.

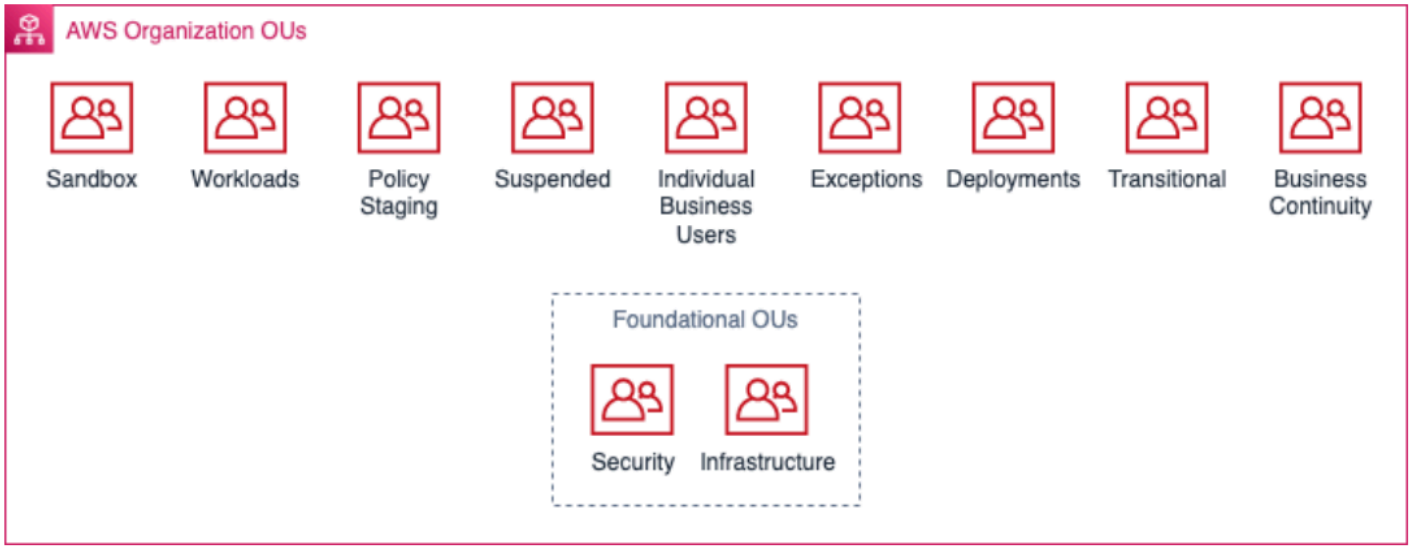
Note

If you are currently using AWS Identity Center and you are not using an external IdP (you are using the IAM Identity Center store or your domain service for your identity source), you can use this break glass access in case of Identity Center failure. Review how to set up emergency access for your IAM Identity Center.
We strongly recommend configuring these users with a hardware-based MFA device, which can be used in exceptional circumstances to gain access to the organization management account or sub-accounts within the organization by assuming a role. While we recommend the use of the organization management account for break glass access, some

organizations might choose to add a dedicated break glass account. This does not eliminate the need for organizational break glass users in the organization management account.

Recommended OUs and accounts

This section provides details on the recommended OUs and, when applicable, a set of recommended AWS accounts.



Recommended OUs

Depending on your requirements, you might not need to establish all the recommended OUs. As you adopt AWS and learn more about your needs, you can expand the overall set of OUs. Refer to the Patterns for organizing your AWS accounts for examples of how you might begin to organize your AWS accounts.

While the provided OU recommendations are geared towards common use cases, it is your organization's responsibility to define a customized OU structure that aligns with your distinct requirements relevant to isolation and automation.

The recommended OUs consist of:

Foundational organizational units (OUs)

Foundational OUs are used to group AWS accounts that support the management, governance, and common infrastructure of your AWS environment.

- **Security OU:** Groups AWS accounts that apply security policies, governance and compliance controls across the organization.

- **Infrastructure OU:** Groups AWS accounts that host and manage core infrastructure and networking services and resources that are shared across the organization.

Application OUs

Application OUs are used to group AWS accounts for production and nonproduction workload environments.

- **Workloads OU:** Groups AWS accounts that host the organization's business-specific workloads, including both production and non-production environments.

Experimental OUs

Experimental OUs are used to group accounts for research and development environments.

- **Sandbox OU:** Groups AWS accounts used for experimentation, testing and development activities, typically with limited access to production resources.

Procedural OUs

Procedural OUs are used to group accounts for process driven activities on AWS Accounts.

- **Exceptions OU:** Groups AWS accounts that host workloads requiring specific configurations or policies that deviate from the organization's standard governance model.
- **Transitional OU:** Temporary OU for housing AWS accounts during migration or restructuring processes, ensuring controlled management and gradual integration into the organization's standard governance structure.
- **Suspended OU:** Groups AWS accounts that have been temporarily suspended or deactivated due to security concerns, policy violations or other administrative reasons.
- **Policy Staging OU:** Hosts AWS accounts that are used to test and validate new or modified organizational policies before applying them to production environments.

Advanced OUs

Advanced OUs are used to group accounts for specific advanced use-cases.

- **Individual Business Users OU:** Groups AWS accounts associated with individual employees or business units, ensuring appropriate access controls and compliance with organizational policies.
- **Deployments OU:** Groups accounts that host services and resources used to orchestrate the deployment of applications, services and infrastructure across multiple AWS accounts within an organization.
- **Business Continuity OU:** Houses resources and accounts specifically designed for disaster recovery, backup and ensuring continuous operations of critical business functions across the organization.

Foundational OUs

The Security OU and the Infrastructure OU are categorized as foundational OUs. Foundational OUs are defined as OUs that contain accounts, workloads, and other AWS resources that provide common security and infrastructure capabilities to secure and support your overall AWS environment.

Accounts, workloads, and data residing in the foundational OUs are typically owned by your centralized Cloud Platform or Cloud Engineering teams made up of cross-functional representatives from your Security, Infrastructure, and Operations teams.

The majority of your accounts are contained in the other OUs. These OUs are intended to contain your business-related workloads. They also contain tools and services that support the entire lifecycle of your business-related services and data.

Security OU

The Security OU is a foundational OU. Your security organization should own and manage this OU along with any child OUs and associated accounts.

We recommend that you create the following accounts in the Security OU:

- Log Archive
- Security Tooling (Audit)

> **Note**
>
> A default deployment of AWS Control Tower will create a Log Archive and Audit (also referred to as Security Tooling) accounts.

Depending on your initial requirements, you might not need to establish all of these accounts.

Log Archive account

The Log Archive is an account that acts as a consolidation point for log data that is gathered from all the accounts in the organization and primarily used by your security, operations, audit, and compliance teams. This account contains a centralized storage location for copies of every

account's audit, configuration compliance, and operational logs. It also provides a storage location for any other audit/compliance logs, as well as application/OS logs. For example, in this account, we recommend that you consolidate AWS API access logs recorded in AWS CloudTrail, logs of changes to AWS resources recorded in AWS Config, and other logs that have security implications.

If you use VPC peering between accounts, then you might also benefit from consolidating VPC Flow Logs data in this account. Logs should generally be made directly available for local use by teams working in any account on a shorter-term retention basis. It is common practice to auto-ingest logs from the log archive account into a security information and event management (SIEM) solution.

> **Note**
>
> By utilizing AWS Control Tower for AWS environment management, it automatically enforces best practices, deploying AWS Config and AWS CloudTrail seamlessly across your environment. Their logs are consolidated in an Amazon S3 bucket within the Log Archive account.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation details	AWS Control Tower enabled
Amazon Security Lake	Amazon Security Lake centralizes security data from cloud, on-premises, and custom sources into a data lake that's stored in your account.	No

Services in the Log Archive account

With Amazon Security Lake, you can automatically centralize security data from AWS and third-party sources into a data lake that's stored in your Log Archive account. Review Managing access in this account in the following sections to learn how to grant access to the logs from other accounts in your AWS organization.

Logs should be available within the workload account for use by teams on short-term retention basis. It is common practice to auto-ingest logs from the log archive account into a security information and event management (SIEM) solution.

If you are using AWS Control Tower to manage your overall AWS environment, then AWS Config is automatically enabled in each Control Tower enrolled account, and AWS CloudTrail Org trail is created for all accounts in the Organization. The AWS CloudTrail logs and AWS Config configuration history are consolidated in an Amazon S3 bucket in the log archive account.

Operational log data

Operational log data used by your infrastructure, operations, and workload owning teams often overlaps with the log data used by security, audit, and compliance teams. We recommend that you consolidate your operational log data into the Log Archive account. Based on your specific security and governance requirements, you might need to filter operational log data saved to this account. You might also need to specify who and what has access to the operational log data in the log archive account.

Immutable log data

Log data housed in the Log Archive account is considered immutable in that it is protected from being changed or deleted. Data retention policies and legislation that apply to your organization might also apply to the data in your log archive account.

Managing access to this account

We strongly recommend that you only house log data in this account. By doing so, access to this account can be greatly limited.

Workloads and tools that need to consume the consolidated log data are typically housed in your other accounts and are granted access through read-only IAM roles to access the log data in a read-only, least privileged manner.

Additionally, to help ensure that log data is properly protected, we recommend SCPs be applied to the Security OU preventing modification or deletion of files within the centralized logging S3 bucket(s).

Additionally, the use of S3 bucket versioning provides visibility into the complete history of all log files.

Security Tooling (Audit) account

> **Note**
>
> In the context of AWS services, this account is used to provide centralized delegated admin access to AWS security tooling and consoles, as well as provide view-only access for investigative purposes into all accounts in the organization. The security tooling account should be restricted to authorized security and compliance personnel and related security. This account is an aggregation point (or points for organizations that split the functionality across multiple accounts) for AWS security services, including AWS Security Hub, Amazon GuardDuty, Amazon Macie, AWS AppConfig, AWS Firewall Manager, Amazon Detective, Amazon Inspector, and IAM Access Analyzer.

`ViewOnlyAccess` and `ReadOnlyAccess` IAM managed policies provide permissions that do not include mutable actions. The `ReadOnlyAccess` grants read access to all AWS services and resources whereas the `ViewOnlyAccess` access provides read-only access and further restricts read operations to view resources and only metadata.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation details	AWS Control Tower enabled
AWS Audit Manager	Continuously audit your AWS use across multiple-accounts in your organization to simplify how you assess risk and compliance. Recommend ed to be in same AWS account AWS Security Hub delegated admin exists. Delegation needs to be done on home and operational AWS Regions.	No
AWS CloudFormation Stacksets	CloudFormation Stacksets can be delegated to multiple	Yes, delegation not configure d

AWS service	Implementation details	AWS Control Tower enabled
	accounts within your AWS Organization. Delegation of the service needs to be completed at only one AWS region for the AWS account.	
AWS CloudTrail	The management of CloudTrail Org Trails can be delegated to one account. It is recommended that the Security team manages the implementation.	Yes, delegation not configure d
AWS Config	Organization-wide aggregate d view of your AWS resources , your AWS Config rules, and the AWS resources' compliance state. Creating an Organization aggregator can be done across multiple AWS regions into the region the aggregator is being deployed to. Multiple accounts can be delegated the AWS Config aggregator.	Yes, delegation not configure d

AWS service	Implementation details	AWS Control Tower enabled
AWS Detective	Required to be deployed to same account which is managing Amazon GuardDuty and AWS Security Hub. Requires GuardDuty to be enabled on Security Tooling account prior to delegating AWS Detective. Delegation needs to be done on home and operational AWS Regions.	No
AWS Firewall Manager	Configure full delegated administration support for Security Tooling account. Firewall Manager delegatio n is a global configuration for all AWS Regions and only needs to be delegated from your home AWS Region.	No
Amazon GuardDuty	Amazon GuardDuty allows for one delegated admin per AWS Organization. It is recommended to delegated Amazon GuardDuty to the same account AWS Security Hub and Amazon Macie are delegated to. Delegation needs to be done on home and operational AWS Regions.	No

AWS service	Implementation details	AWS Control Tower enabled
Amazon Inspector	Delegate an administrator to enable or disable scans for member accounts, view aggregated finding data from the entire organization, create and manage suppression rules. Delegation needs to be done on home and operation al AWS Regions.	No
Amazon Macie	Amazon Macie allows for one delegated admin per AWS Organization. It is recommended to delegated Amazon Macie to the same account AWS Security Hub and Amazon GuardDuty are delegated to. Delegation needs to be done on home and operational AWS Regions.	No
AWS Security Hub	AWS Security Hub allows for one delegated admin per AWS Organization. It is recommended to delegated AWS Security Hub to the same account Amazon GuardDuty and Amazon Macie are delegated to, for ease of pivoting between these services in the AWS Management Console. Delegation needs to be done on each operational Region.	Yes — When you activate a Security Hub detective control within AWS Control Tower, it automatically enables Security Hub on your behalf.

AWS service	Implementation details	AWS Control Tower enabled
Amazon S3 Storage Lens	Allows for multiple delegated admin accounts per AWS Organization. Service is global and only needs to be delegated from the home AWS Region	No
AWS Trusted Advisor	Allows for centralized view of AWS Trusted Advisor information. Requires the management account in your organization must have a Business, Enterprise On-Ramp, or Enterprise Support plan. Service is global and only needs to be delegated from the home AWS Region.	No
IAM Access Analyzer	Configured with the entire AWS organization as the zone of trust so that it's easier for you to quickly look across resource policies and identify resources with public or cross-account access you might not intend. We recommend that you configure this analyzer in one of your security tooling accounts.	No

Additional services and functionalities

Common examples of security capabilities that can be centrally accessed and managed using the Security Tooling account include:

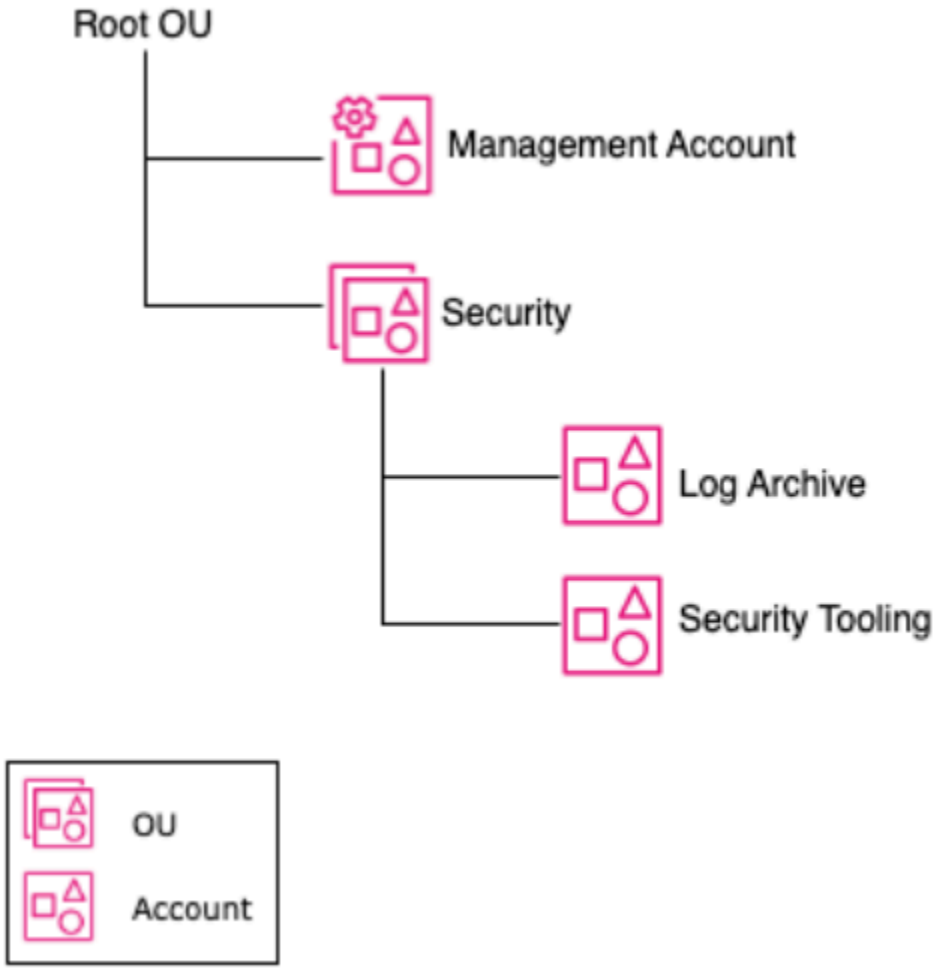
- **Third-party cloud security monitoring tools** — You can also house third-party cloud security monitoring services and tools in your security tooling accounts. For example, these accounts typically contain security information and event management (SIEM) tools and vulnerability scanners
- **Automated detection and response workflows** — Automated detection and response workflows that act on data collected through these types of services are normally contained in your security tooling accounts.
- **Incident response (IR) support** — Tools to support manual incident response (IR) procedures are typically housed in your security tooling accounts. Refer to the AWS Security Incident Response Guide for more information.

AWS Solutions

AWS Solution	Description
Automated Security Response on AWS	Add-on that works with AWS Security Hub and provides predefined response and remediati on actions based on industry compliance standards and best practices for security threats. It helps Security Hub customers to resolve common security findings and to improve their security posture in AWS.
Automations for AWS Firewall Manager	Allows you to centrally configure, manage, and audit firewall rules across all your accounts and resources in AWS Organizations. This solution is a reference implementation to automate the process to set up AWS Firewall Manager security policies.
Security Automations for AWS WAF	Automatically deploys a set of AWS WAF (web application firewall) rules that filter common web-based attacks. Users can select from preconfigured protective features that define the rules included in an AWS WAF web access control list (web ACL).

Example structure

The following example structure represents the recommended Security OU at a basic level. Note that within Control Tower governed environments, the accounts within the Security OU are limited to the Log Archive and Security Tooling (also known as Audit by default for AWS Control Tower deployments).



Example structure of Security OU

Infrastructure OU

The Infrastructure OU is a foundational OU that is intended to contain infrastructure services. The accounts in this OU are also considered administrative and your infrastructure and operations teams should own and manage this OU, any child OUs, and associated accounts.

The Infrastructure OU is used to hold AWS accounts containing AWS infrastructure resources that are shared, utilized by, or used to manage accounts in the organization. This includes centralized operations or monitoring of your organization. No application accounts or application workloads are intended to exist within this OU.

Common use cases for this OU include accounts to centralize management of resources. For example, a Network account might be used to centralize your AWS network, or an Operations Tooling account to centralize your operational tooling.

> **Note**
>
> For guidance on where to contain non-infrastructure shared services, refer to Workloads OU.

In most cases, given the way most AWS Organization integrated services interact with the accounts within the Infrastructure OU, it does not generally make sense to have production and non-production variants of these accounts within the Infrastructure OU. In situations where non-production accounts are required, these workloads should be treated like any other application and placed in an account within the appropriate Workloads OU corresponding with the non-production phase of the SDLC (Dev OU or Test OU).

Backup account

The Backup account serves as a dedicated and centralized hub for backup and disaster recovery management. It provides a unified platform to orchestrate, monitor, and enforce backup policies across AWS accounts within the AWS Organization.

By consolidating backup processes in a central account, organizations can achieve several benefits. It simplifies backup management by eliminating the need to configure and maintain backup settings separately in each member account, streamlining operational efficiency and reducing the potential for errors. It ensures consistent and comprehensive data protection across the entire AWS infrastructure, regardless of the specific AWS services and resources in use. This approach also enhances compliance and governance efforts by enabling centralized auditing and reporting on backup and recovery activities, making it easier to track data protection metrics and maintain necessary records for compliance purposes.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation details	AWS Control Tower enabled
AWS Backup	Register the Backup account as the delegated administrator in the AWS Backup console.	Yes
AWS Organizations: AWS Backup policy administration	Delegate AWS Backup Policy administration to the Backup account by enabling delegation of AWS Organizations in the management account and configure a policy that allows the Backup account to create Backup Policies.	Yes

Additional services and functionalities

Common examples of security capabilities that can be centrally accessed and managed using the Backup account includes:

- Use centralized AWS KMS customer managed keys for AWS Backup service within the Backup account to centrally manage the encryption for backup operations across accounts.
- Third-party backup tools that require resources can be created and managed in the Backup account.

Identity account

The Identity account serves as a centralized identity federation account isolated from all other management and workload activities within the AWS Organization. Federated identity management grants you the ability to efficiently manage the access to the accounts in the AWS Organization and authorization to integrated applications. By managing your identities and controlling access to your environment centrally, you can quickly create, update, and delete the permissions and policies you need to meet your business requirements.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation details	AWS Control Tower enabled
IAM Identity Center	You can delegate administration of IAM Identity Center to this account which will allow you to administer IAM Identity Center outside of the management account.	Enabled — Yes Delegated — No
IAM Access Analyzer	An IAM Access Analyzer can be configured to detect resources that are shared outside of the organization (organization zone of trust). By default, this is managed from the management account. This can be delegated to a member account. This can be delegated to the Identity account or a Security Tooling account depending on who is responsible for auditing external access (Identity Team or Security Team).	No
Policy management for Organizations	From the organization's management account, you can delegate policy management for Organizations to specified member accounts to perform policy actions that are by default available only to the management account.	No

AWS service	Implementation details	AWS Control Tower enabled
Central management root access for member accounts	We recommend you centrally secure the root user credentials of AWS accounts managed using AWS Organizations to prevent root user credential recovery and access at scale.	No

Additional services and functionalities

Common examples of security capabilities that can be centrally accessed and managed using the Identity account includes:

- **AWS Directory Service** — If you are using an AWS-hosted directory or AWS AD Connector, you can create and managed them in your Identity account alongside of AWS IAM Identity Center.
- **SAML 2.0 custom managed applications** — With IAM Identity Center, you can create or connect workforce users and centrally manage their access across all their AWS accounts and applications.

Network account

The Network account serves as the central hub for your network within your AWS Organization. You can manage your networking resources and route traffic between accounts in your environment, your on-premises, and egress/ingress traffic to the internet. Within this account, your network administrators can manage and build security measures to protect network traffic across your cloud environment.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation details	AWS Control Tower enabled
AWS Network Manager	Centrally manage and monitor your global networks with transit gateways and their attached resources in	No

AWS service	Implementation details	AWS Control Tower enabled
	multiple AWS accounts within your organization.	
IPAM	Delegated to a single account for your entire AWS Organization. IPAM will inventory and track all active IPs across your AWS Organization.	No
VPC Reachability Analyzer	Trace paths across accounts in your organizations. You can assign multiple delegated admin accounts as needed.	No

Additional services and functionalities

Common examples of network capabilities and AWS services that can be centrally accessed and managed via the Network account include:

- **Amazon VPC** — If you plan to implement centralized networking in your AWS environment, we recommend managing your VPCs within your network account, and sharing resources across your accounts within your AWS organization.
- **Share your AWS Transit Gateway** — Create an AWS Transit Gateway resource in the networking account and share it across the accounts within your AWS Organization using AWS Resource Access Manager (RAM).
- **Share your Amazon Route 53 Endpoint Resolvers** — If you plan to use a centralized transitive network with Amazon Route 53 Public Data Plane in your AWS Organization, we recommend managing and sharing your Route 53 Endpoint Resolvers in your network account within your AWS organization.
- **Share your IPAM pools with your organization** — When you delegate an IPAM account, IPAM enables other AWS Organizations member accounts in the organization to allocate CIDRs from IPAM pools that are shared using AWS Resource Access Manager (RAM).

- **Build centralize AWS Site-to-Site VPN connections** — Using a transitive network architecture centralized in your Network account, a site-to-site VPN can be established and routing enabled across your cloud environment.
- **Centralize AWS Direct Connect** — Create and attach AWS Direct Connect to your transitive network with AWS Transit Gateway.
- **Centralized network inspection point** — Build inbound and outbound network traffic inspection points routing through the Network account.

AWS Solutions

The following AWS Solutions are commonly deployed or related to the functional operations of the Network account:

AWS Solution	Description
Network Orchestration for AWS Transit Gateway	Automates the process of setting up and managing transit networks in distributed AWS environments. This solution allows customers to visualize and monitor their global network from a single dashboard rather than toggling between Regions from the AWS console. It creates a web interface to help control, audit, and approve transit network changes.
Automations for AWS Firewall Manager	Allows you to centrally configure, manage, and audit firewall rules across all your accounts and resources in AWS Organizations. This solution is a reference implementation to automate the process to set up AWS Firewall Manager security policies.
Security Automations for AWS WAF	Automatically deploys a set of AWS WAF (web application firewall) rules that filter common web-based attacks. Users can select from preconfigured protective features that define

AWS Solution	Description
	the rules included in an AWS WAF web access control list (web ACL).

Operations Tooling account

Operations Tooling accounts can be used for day-to-day operational activities across your organization. The operations tooling account hosts tools, dashboards, and services needed to centralize operations where monitoring and metric tracking are hosted. These tools help the central operations team to interact with their environment from a central location.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation details	AWS Control Tower enabled
AWS Account Management	Manage alternate contact information for all of the accounts in your organization. Delegation is done on one region and for one account within your AWS Organizations.	No
AWS Application Migration Service (AMG)	AWS Application Migration Service simplifies, expedites, and reduces the cost of migrating applications to AWS. By integrating with Organizations, you can use the global view feature to manage large-scale migrations across multiple accounts.	No
Amazon DevOps Guru	You can integrate with AWS Organizations to manage insights from all	No

AWS service	Implementation details	AWS Control Tower enabled
	accounts across your entire organization. You delegate an administrator to view, sort, and filter insights from all accounts to obtain organization-wide health of all monitored applications.	
AWS Health	Get visibility into events that might affect your resource performance or availability issues for AWS services. You can register up to 5 member accounts in your organization as a delegated administrator.	No
AWS License Manager	If you are planning to use a centralized model to buy and share licenses across your organization, we recommend you specify one of your Shared Services accounts as the delegated administrator for AWS License Manager.	No
AWS Systems Manager Change Manager	You can delegate administration for Systems Manager to the Operations Tooling account to perform administrative tasks for Change Manager, Explorer, and Ops Center.	No
AWS Systems Manager Explorer		No

AWS service	Implementation details	AWS Control Tower enabled
AWS CloudFormation Stacksets	You can register multiple delegated administrator accounts in your AWS Organizations. CloudFormation Stackset delegation will give the AWS account full administrative access to deploy resources in other AWS accounts in your Organization. Delegation needs to be done only at the home region.	No
VPC Reachability Analyzer	Trace paths across accounts in your organizations. VPC Reachability Analyzer can have multiple delegated admin accounts.	No

AWS Solutions

The following AWS Solutions are commonly deployed or related to the functional operations of the Operations Tooling account:

AWS Solution	Description
Account Assessment for AWS Organizations	Presented in a web UI, this AWS Solution runs configurable scans on all AWS accounts in your AWS Organizations to help you identify dependencies in your underlying resource-based policies.
Instance Scheduler on AWS	Automates the starting and stopping of Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Relational Database Service

AWS Solution	Description
	(Amazon RDS) instances. This solution helps reduce operational costs by stopping resources that are not in use and starting them when they are needed. The cost savings can be significant if you leave all of your instances running at full utilization continuously.
Cost Optimizer for Amazon WorkSpaces	Analyzes all of your Amazon WorkSpaces usage data and automatically converts the WorkSpace to the most cost-effective billing option (hourly or monthly), depending on your individual usage. You can use this solution with a single account, or with AWS Organizat ions across multiple accounts, to help you monitor your WorkSpace usage and optimize costs.
Workload Discovery on AWS	Workload Discovery on AWS (formerly called Amazon Personalize) is a tool to visualize AWS Cloud workloads. Use Workload Discovery on AWS to build, customize, and share detailed architecture diagrams of your workloads based on live data from AWS.

Monitoring account

An AWS monitoring account can be used to monitor resources, applications, log data, and performance in other AWS accounts. AWS offers a number of tools and services that can be used to manage and monitor resources and workloads in an AWS account, including CloudWatch, Amazon Managed Service for Prometheus, Amazon Managed Grafana, and Amazon OpenSearch Service. These tools can be used to monitor resource and application usage, performance, review log data, and identify potential issues within the infrastructure or application.

Note

Depending on your business requirements and team structures, you may choose to manage your monitoring resources and services in a single account with your other Operational Tooling services or as a dedicated Monitoring account. The core concept of the Monitoring account is to only give read-only functionality. The account in itself is not intended to have the ability to make changes across account your AWS Organization.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation details	AWS Control Tower enabled
AWS Health	Configure the Monitoring account as the delegated admin for AWS health (in the Management account) for ongoing visibility into your resource performance and the availability of your AWS services and accounts within your organization.	No
Amazon S3 Storage Lens	Register the Monitoring account as the delegated admin for Amazon S3 storage Lens (in the Management account) for organization-wide visibility into object-st orage usage and activity. You can use S3 Storage Lens metrics to generate summary insights, such as finding out how much storage you have across your entire organizat	No

AWS service	Implementation details	AWS Control Tower enabled
	ion or which are the fastest-g rowing buckets and prefixes.	

Additional services and functionalities

Common examples of monitoring capabilities that can be centrally accessed and managed using the Monitoring account includes:

- **AWS CloudWatch** — Configure AWS CloudWatch Cross Account observability and configure as the *monitoring account* or hub account.
- **CloudWatch dashboards** that are created at the account level can be shared with the monitoring account which allows for distributed management with centralized monitoring.
- **Third-party monitoring tools** (such as ElasticSearch, Splunk, Prometheus, and Grafana) that require resources can be created and managed in the Monitoring account.
- **Customer created automations and reports** can be run from and stored in the Monitoring account.
- **Log Archive log analysis**. In order to analyze Log data stored in the Log Archive account, Amazon Managed Grafana or QuickSight can be used in the Monitoring account to analyze Log data in an S3 bucket in the Log Archive account by connecting to Amazon Athena in the Log Archive account.
- **Amazon OpenSearch Service** can be deployed and managed in the Monitoring account to analyze logs, monitor applications, and analyze clickstreams.
- **QuickSight** can be deployed and managed in the Monitoring account and cross account data sources can be used to centrally monitor or report organization data.
- **Amazon Managed Grafana** can be deployed into the monitoring account for centralized monitoring of resources, containers, CloudWatch logs, and applications by connecting to data sources in different accounts or to centralized CloudWatch metrics, logs, and traces.

AWS Solutions

The following AWS solutions are commonly deployed or related to the functional operations of the Monitoring account:

AWS Solution	Description
Centralized Logging with OpenSearch	Helps organizations collect, ingest, and visualize log data from various sources using Amazon OpenSearch Service. This solution provides a web-based console, which you can use to create log ingestion pipelines with a few clicks.

Shared Services accounts

A Shared Services account is an AWS account created and dedicated to hosting and managing centralized IT services and resources that are shared across multiple other AWS accounts within an AWS Organization. The primary purpose of a Shared Services account is to consolidate similar shared services to give a single access point to manage, interface and consume. You may create multiple Shared Service accounts depending on your need to securely isolate the functionality of the grouped services in the account.

> **Note**
>
> AWS account workload isolation is a best practice for enhancing security and operational efficiency in cloud environments. It involves grouping AWS resources and workloads into separate AWS accounts based on their functionality and security requirements. A Shared Service account should contain resources and workloads that can be grouped together in order to ensure security, compliance, and operational separation of duties.

Recommended AWS Organization Integrated Service Delegation

AWS service	Implementation Details	Control Tower Enabled
Service Catalog	Create and manage catalogs of IT services that are approved for use on AWS.	Yes — AWS Control Tower automatically sets up Service Catalog to provision new accounts through Account Factory.

AWS service	Implementation Details	Control Tower Enabled
AWS Compute Optimizer	AWS Compute Optimizer can be delegated to one AWS account in your AWS Organization. It is recommend ed to deploy to a Shared Services account or the Monitoring account.	No

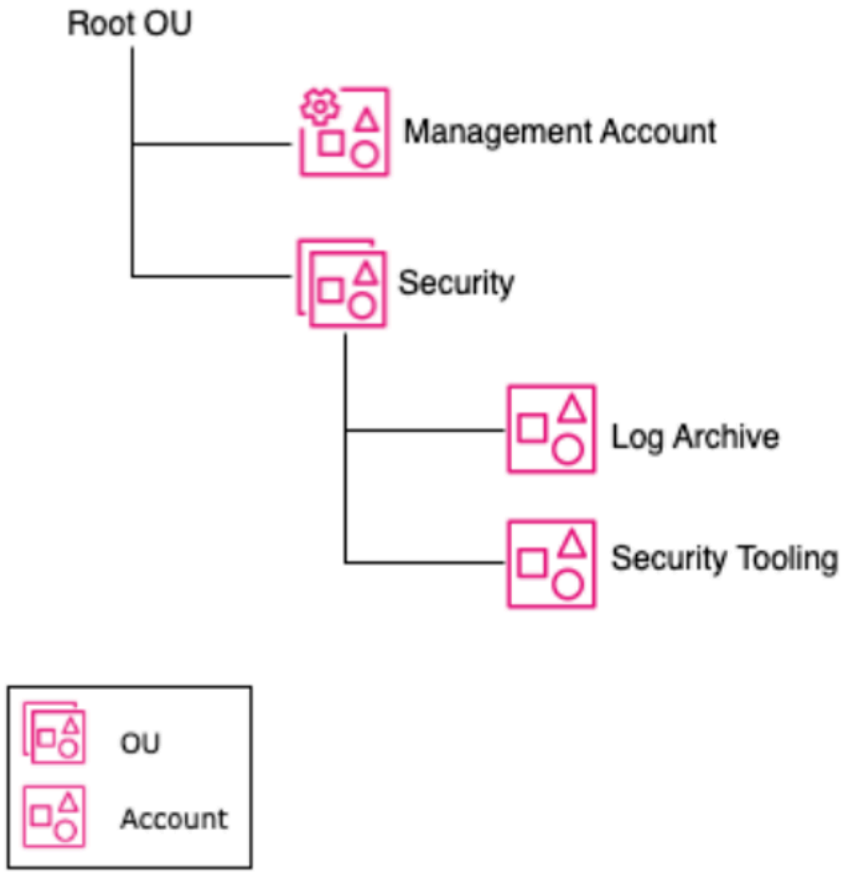
Additional services and functionalities

Common examples of security capabilities that can be centrally accessed and managed using the Shared Services account includes:

- **EC2 Image Builder** — EC2 Image Builder integrates with AWS Resource Access Manager (AWS RAM) to allow you to share certain resources with any AWS account or through AWS Organizations.

Example structure

The following example structure represents the recommended Infrastructure OU at a basic level. For general guidance on separating production and non-production workloads, refer to Organizing workload-oriented OUs.



Example structure of Infrastructure OU

Application OUs

Workloads OU

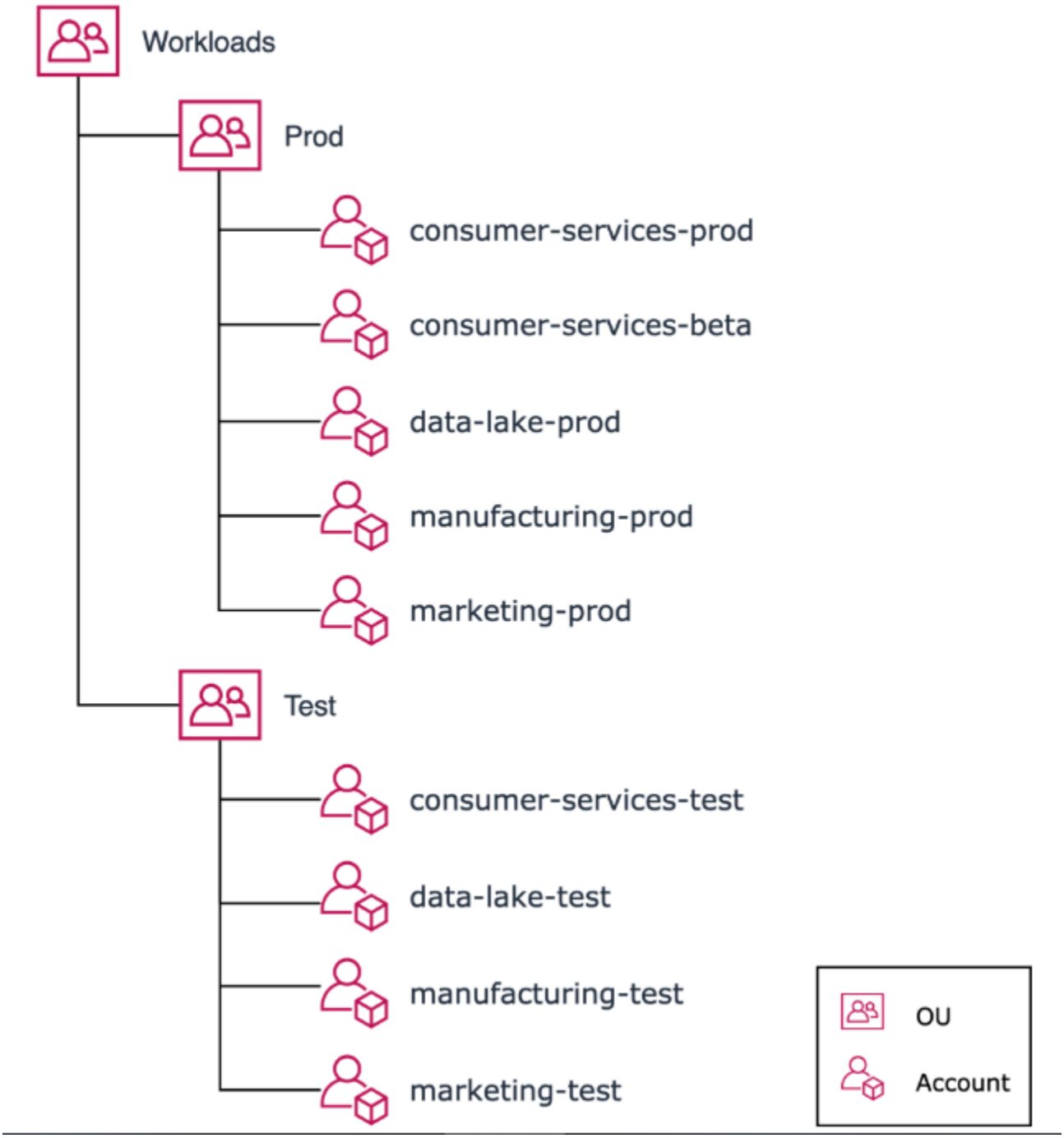
The Workloads OU is intended to house most of your business-specific workloads including both production and non-production environments. These workloads can be a mix of commercial off-the-shelf (COTS) applications and your own internally developed custom applications and data services.

Workloads in this OU often include shared application and data services that are used by other workloads.

Example structure

The following example represents a basic structure in which sets of workloads owned by diverse business units or teams reside in two child OUs: Prod and Test. In this example, a common governance and operating model applies across those areas. The data-lake-prod account shown in this example contains data services that are shared with other production workloads and accounts.

For general guidance on separating production and non-production workloads and resources, refer to Organizing workload-oriented OUs.



Example structure of Workloads OU

Experimental OUs

Sandbox OU

The Sandbox OU contains accounts in which your builders are generally free to explore and experiment with AWS services and other tools and services subject to your acceptable use policies. These environments are typically disconnected from your internal networks and internal services. Sandbox accounts should not be promoted to any other type of account or environment within the Workloads OU.

Sandbox per builder or team with spend limits

A common practice is to provide a sandbox account to either each builder or each small team of builders, along with cloud spend budgets to ensure that their AWS spending aligns within your policies. In more advanced scenarios, you might provide your builders and teams with the option to have multiple sandbox accounts so that they can experiment more freely with configurations that entail use of multiple accounts (for example, experimenting with cross-account IAM roles).

There is a maximum number of accounts in an organization. If you have thousands of builders and expect to allocate a sandbox environment for each builder, you might encounter the maximum quota for accounts. Refer to Quotas for AWS Organizations for more details on the maximum number of accounts in an organization.

In cases where you need more than several thousand sandbox accounts, you might consider either creating one or more separate organizations to contain the sandbox accounts or establishing a process to recycle sandboxes when they are no longer in use.

Temporary resources and environments

Unlike more persistent development environments, it's common to set expectations with your builders that the resources they create in sandbox environments are temporary in nature. As a cost control measure and to reinforce the temporary nature of sandbox resources, you can put automated procedures in place to periodically purge the resources created in these environments.

As a further measure to reduce costs, you can use the Instance Scheduler on AWS solution to automate the starting and stopping of Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Relational Database Service (Amazon RDS) instances based on provided schedules

Wide-ranging access

Wide-ranging access is typically provided in sandbox-oriented accounts including administrative-like access within each account, full access to most AWS services, and possibly outbound and inbound access to the internet. Access to the internet might be required to connect to AWS service APIs, download externally accessible software packages, and integrate with publicly available services.

No access to corporate resources and non-public data

Given the extent of access provided in sandbox environments, businesses typically employ a combination of controls and internal usage agreements to limit builders from accessing corporate resources and data from their sandbox accounts. Use of non-public data and intellectual property, including proprietary source code and binaries, is typically not allowed in sandbox environments.

Sandbox and development environments

Due to use of non-public data and the more formal nature of the work being performed in development environments, we recommend that you make a high-level distinction between sandbox environments and development environments. For example, in development environments your teams are performing more formal experiments, day-to-day development, and early testing work that requires access to your intellectual property and to enterprise services, such as source code and artifact management.

Additional services and functionalities

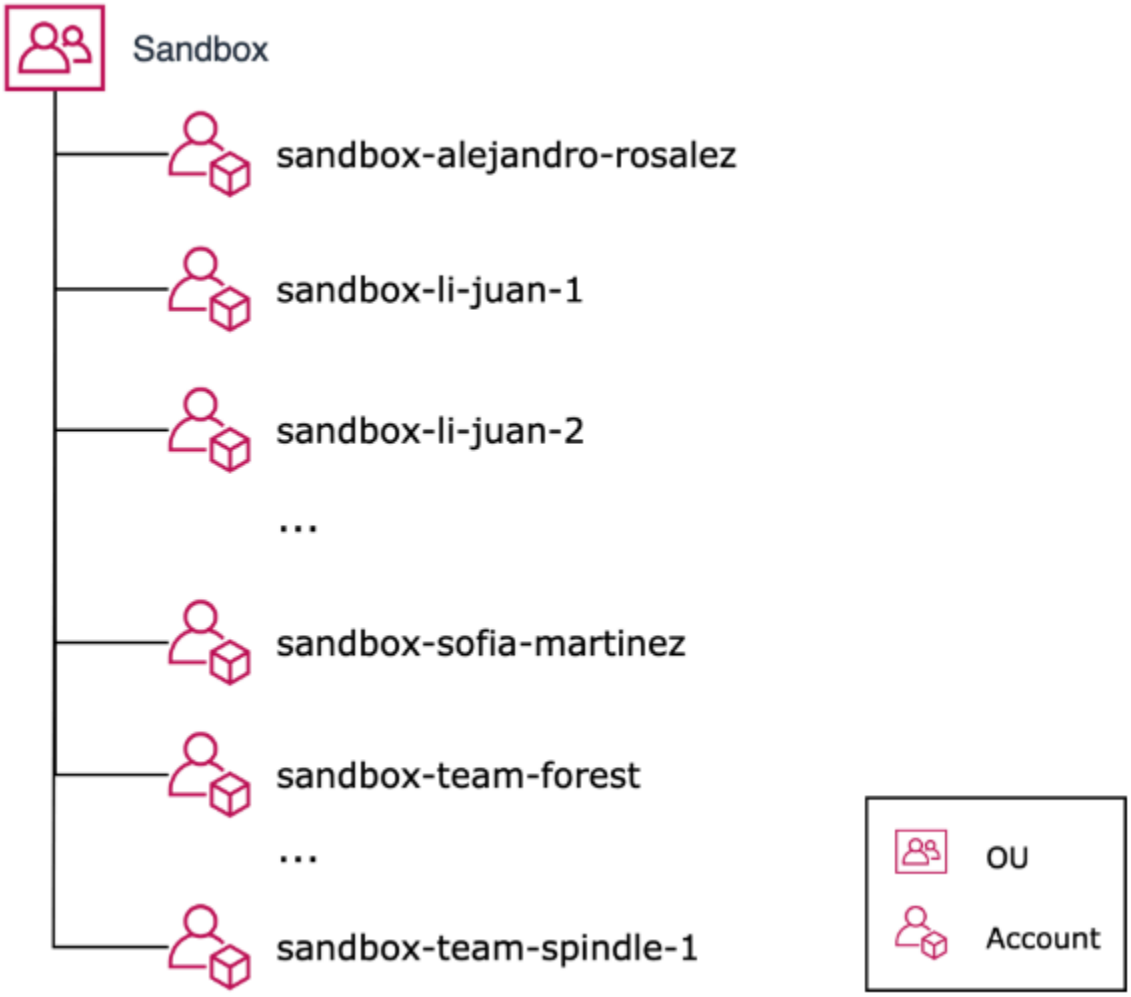
Common examples of monitoring capabilities that can be centrally accessed and managed using a Shared Services account include:

- **Instance Scheduler on AWS** solution to automate the starting and stopping of Amazon Elastic Compute Cloud (Amazon EC2) and Amazon Relational Database Service (Amazon RDS) instances based on provided schedules.

Sandbox per builder or team

In the following example, sandbox accounts are represented for individual builders and teams. One user has two sandbox accounts so that they can perform experiments that require multiple accounts.

In support of hackathons and other events, you might also find value in creating transient sandbox accounts for temporary teams of people.



Example structure of Sandbox OU

Procedural OUs

Exceptions OU

The Exceptions OU houses accounts that require an exception to the security policies that are applied to your Workloads OU. Normally, there should be a minimal number of accounts, if any, in this OU.

Service control policies and scrutiny

Given the unique nature of the exceptions, SCPs are typically applied at the account level in this OU. Due to the customized security controls that apply to these accounts, owners of these accounts can expect to experience greater scrutiny from security monitoring systems.

Consider Workloads OU as an alternative

If you observe a pattern in which multiple accounts require the same set of exceptions, we recommend that you examine either your existing workloads policies or an extended form of the Workloads OU structure and house the accounts under the Workloads OU. You can introduce another level of OU under the Workloads OU to represent a common set of security policies and/or operational processes that can be applied to multiple workload environments. For more information, refer to Organizing workload-oriented OUs.

Transitional OU

The Transitional OU is intended as a temporary holding area for existing accounts and workloads that you move to your organization before formally integrating them into the more standardized areas of your AWS environment structure.

Common scenarios for moving accounts into your organization

Common scenarios for moving accounts into your organization include:

- Acquisition of a company that is already using AWS and has a set of accounts
- Existence of your own accounts that were created before you established your newer AWS environment structure

- Movement of accounts that have previously been managed by a third party
- Divestment of specific workload to be migrated out of your AWS Organization

Considerations for moving accounts into your organization

If you plan to move an account from an existing organization, you must first remove the account from the organization. For more information, refer to Removing a member account from your organization. Once an account is removed from an organization, it is referred to as a standalone account.

Moving a standalone account that does not have dependencies on other accounts is a straightforward process. In this case, there's generally no need to migrate or modify the existing workloads in the account to be moved. For more information, refer to Inviting an account to join your organization.

If the standalone account to be moved has dependencies on other accounts, then you should evaluate those dependencies to determine if they should be addressed before moving the account.

In your target organization, we recommend that you review SCPs in the organization's root to ensure that those SCPs won't adversely impact the accounts to be moved.

If you're moving a set of related accounts to your organization, you can create a child OU under the Transitional OU for the related set of accounts.

After moving accounts

Over time, as you better understand the direction for these accounts and the workloads contained in them, you can either move the accounts to your Workloads OU as is, invest in migrating the workloads to other accounts, or decommission either the workloads or accounts.

Policy Staging OU

The Policy Staging OU is intended to help teams that manage overall policies for your AWS environment to safely test potentially broadly impacting policy changes before applying them to the intended OUs or accounts. For example, SCPs and tag policies should be tested prior to applying them to the intended OUs or accounts. Similarly, broadly applicable account baseline IAM roles and policies should also be tested using the Policy Staging OU.

Workload-specific policies

Development and testing of workload-specific IAM roles and policies do not need to use the Policy Staging OU. Rather, workload owning teams typically develop and test these resources alongside other workload-specific resources in development and test accounts within your Security, Infrastructure, and Workloads OUs.

Recommended testing and promotion workflow

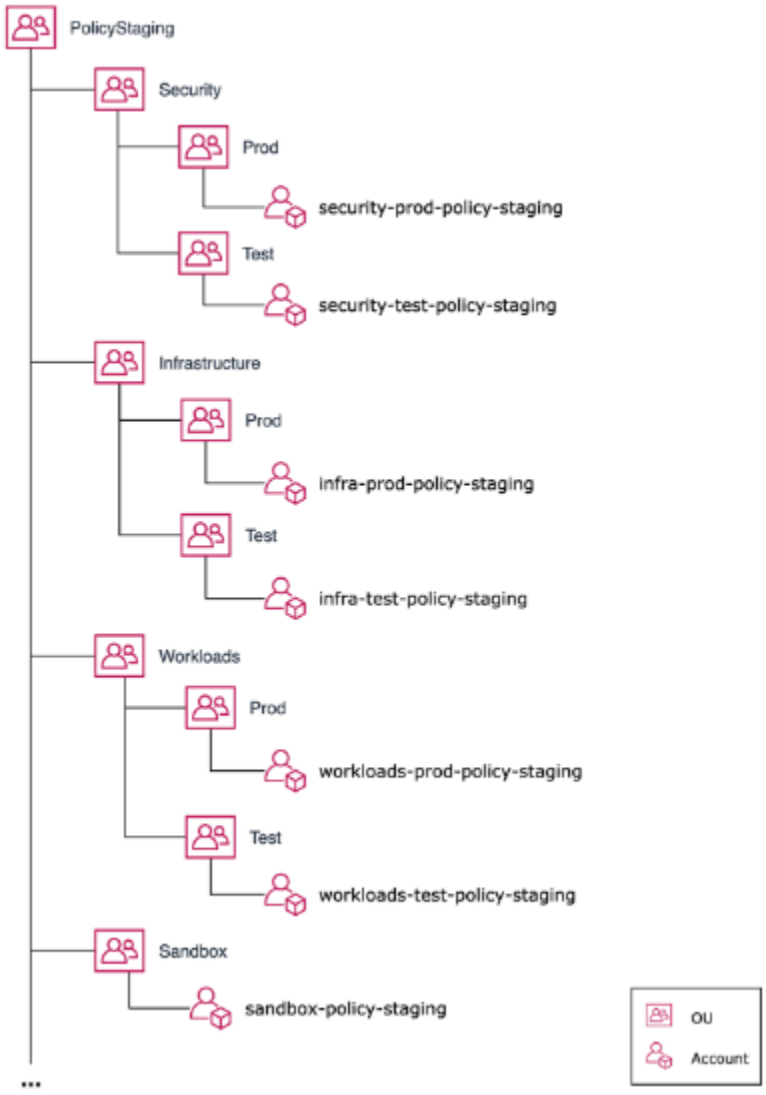
Once you have tested changes in the Policy Staging OU, we recommend that you temporarily associate the policy changes to a single account in the intended OU. If the changes are ultimately targeting an OU, apply the changes to the intended OU and remove the changes from the account only after you have validated that the changes are working as intended.

This approach enables you to validate the changes in production before more broadly applying them.

Example structure

In this example, a set of child OUs mirrors an overall OU structure. At least one test account is included under each child OU.

In support of testing SCPs and tag policies that are intended to be applied at the OU level, your teams should first apply them to one of the test child OUs. SCPs and tag policies that are applied to a specific account require creation of a test account under the appropriate test child OU.



Example structure of Policy Staging OU

Suspended OU

The Suspended OU is used as a temporary holding area for accounts that are required to have their use suspended.

Moving an account to this OU does not automatically change the overall status of the account. For example, in cases where you intend to permanently stop using an account, you would follow the Closing an account process to permanently close the account.

Examples of using the Suspended OU include:

- A person's sandbox account is no longer needed due to the departure of the person from the company.
- A workload account is no longer needed due to the resources having been either retired or migrated to another account.

Constraining activity in suspended accounts

You can use service control policies (SCPs) to inhibit users other than your security and cloud platform teams from using AWS APIs in each account. Additionally, you can remove application-level access so that users can no longer access and manage application resources for each suspended account.

To reduce risk and potentially minimize costs, you can also stop any running resources and applications in each suspended account.

Resources should not be deleted from a suspended account unless the account is intended to be closed.

Tagging suspended accounts

Because you might use the Suspended OU for a variety of use cases, we recommend that you apply tags to each account to record the reason for moving the account and the OU from where the account originated. Each process that you establish to support your suspension use cases can use the tag to automatically process the suspended accounts. This tag can also aid in your internal tracing and auditing of an account's lifecycle.

Closing suspended accounts

If an account is moved to this OU prior to the start of the closure process, you can implement a policy and process to automatically start the account closing process a certain number of days after an account has been moved to this OU.

Once the account closure process has been completed, the account is no longer visible in your organization.

Advanced OUs

Individual Business Users OU

The Individual Business Users OU houses accounts for individual business users and teams who need access to directly manage AWS resources outside the context of resources managed within your Workloads OU.

In some cases, you can consider a small number of AWS resources as something other than a workload. For example, a business team might require write access to Amazon S3 buckets to share marketing videos and data with a business partner. In these cases, you might choose to manage these resources in accounts within the individual business users OU rather than in accounts in the Workloads OU.

Controls

We recommend that you apply a combination of SCPs and IAM permissions to this OU and authorized users. This ensures that only those AWS services, resources, and actions needed are granted. Depending on the nature of the use cases, you can apply controls to individual accounts in this OU.

Services that do not require direct user access to accounts

The individual business users OU does not apply when users can authenticate and be authorized to interact with applications and services without requiring direct access to an account. For example, business users often need access to QuickSight for business intelligence (BI) purposes.

Assuming that you consider your QuickSight-based BI capability a workload, you can position the QuickSight resources and data in a workloads account in the Workloads OU. In this case, BI users are authorized to access the QuickSight service directly without needing access at the account level.

Deployments OU

The Deployments OU contains resources and workloads that support how you build, validate, promote, and release changes to your workloads. You might already be using continuous integration/continuous delivery (CI/CD) capabilities to help manage and automate how changes to various types of source code are processed.

Using CI/CD capabilities residing outside of your AWS environment

If you already use on-premises and/or managed CI/CD and related capabilities that reside outside of your AWS environment and you do not expect to use and/or manage CI/CD services within your AWS environment in the near term, you might not immediately need to establish the Deployments OU and an associated set of CI/CD oriented accounts.

In this scenario, you must work through any access and potential network connectivity dependencies between your CI/CD capabilities residing outside of your AWS environment and your workloads environments in AWS.

Separating CI/CD management capabilities from workloads

If you intend to deploy and/or manage your own CI/CD capabilities in AWS or use AWS managed CI/CD services, we recommend that you use a set of production deployment accounts within the Deployments OU to house the CI/CD management capabilities.

Reasons for separating your CI/CD management capabilities from your workload environments include:

- **Critical roles played by CI/CD capabilities** — Your CI/CD capabilities are responsible for orchestrating quality validation, security compliance checks, building and publishing production candidate artifacts, promoting artifacts, and ultimately triggering release of artifacts to production environment.

  Given the critical nature of these roles, it's important that you can apply appropriate policies and operational practices to your CI/CD capabilities that are different than those applied to your workload environments.

  For example, your CI jobs and CD pipelines typically need write access to publish and promote candidate artifacts to an artifact management service. However, your production workload environments should only require read access to artifact management services in order to obtain the already built and promoted artifacts.
- **CD pipelines affect non-production and production workload environments** – When CD pipelines orchestrate the validation of changes and ultimately trigger the release of changes to production, the pipelines often need to access workloads residing in both non-production test and production workload environments. For example, if you manage your CI/CD capabilities in your production workload environments, then you must allow the production workload environments to access your non-production environments. By centralizing your

CI/CD capabilities in your CI/CD accounts, you can avoid enabling your production workload environments access to non-production environments.

- **CI/CD capabilities depend on unique tooling** – Your CI/CD management capabilities, CI jobs, and CD pipelines often depends on tools that are different from those required to run and operate your workloads. Limiting the use of these tools to your CI/CD accounts can help you reduce the complexity and attack surface of your workload environments.

Running CI jobs and CD build stages in deployment accounts

Because CI jobs and CD pipeline build stages are responsible for generating formal candidate artifacts, we recommend that you perform these activities in a production environment. Rather than perform these activities in your production workload environments, we recommend that you run them in your production CI/CD accounts.

Business Continuity OU

> **Note**
>
> The Business Continuity OU is an advanced use-case topic where your AWS Organization requires data isolation and data residency controls based on unique workloads requirements. In general, most cross account disaster recovery strategies can be implemented through using the Backup account within the Infrastructure OU.

The Business Continuity OU is intended to help teams implement a cross-account disaster recovery strategy. The data is as close to air-gapped as possible and the OU has no workload resources. This creates a secure data bunker to help protect your organization and allow for recovery from severe disasters like ransomware. The secure data bunker should only be accessed when the disaster recovery data for a workload is unavailable, untrustworthy, or destroyed.

The Business Continuity OU does not replace normal disaster recovery plans of your workloads. It's an additional layer of protection that is meant to enhance the resiliency of your organization. General recommendations for disaster recovery for workloads can be found in Disaster Recovery of Workloads on AWS: Recovery in the Cloud.

For organizations that have data residency requirements and are in a geographical area that has only a single AWS Region available, using AWS Outposts can assist in maintaining compliance.

The blog Ensure Workload Resiliency and Comply with Data Residency Requirements with AWS Outposts.

Controls

For the Business Continuity OU to be a secure data bunker, access should be heavily restricted to prevent the data from being compromised. Ideally, users with access to the data within the Business Continuity OU should not have access to the regular environment and users with access to the regular environment should not have access to the Business Continuity OU. Apply a combination of SCPs and IAM permissions to this OU and authorized users to ensure that only those AWS services, resources, and actions needed are granted.

Additional considerations

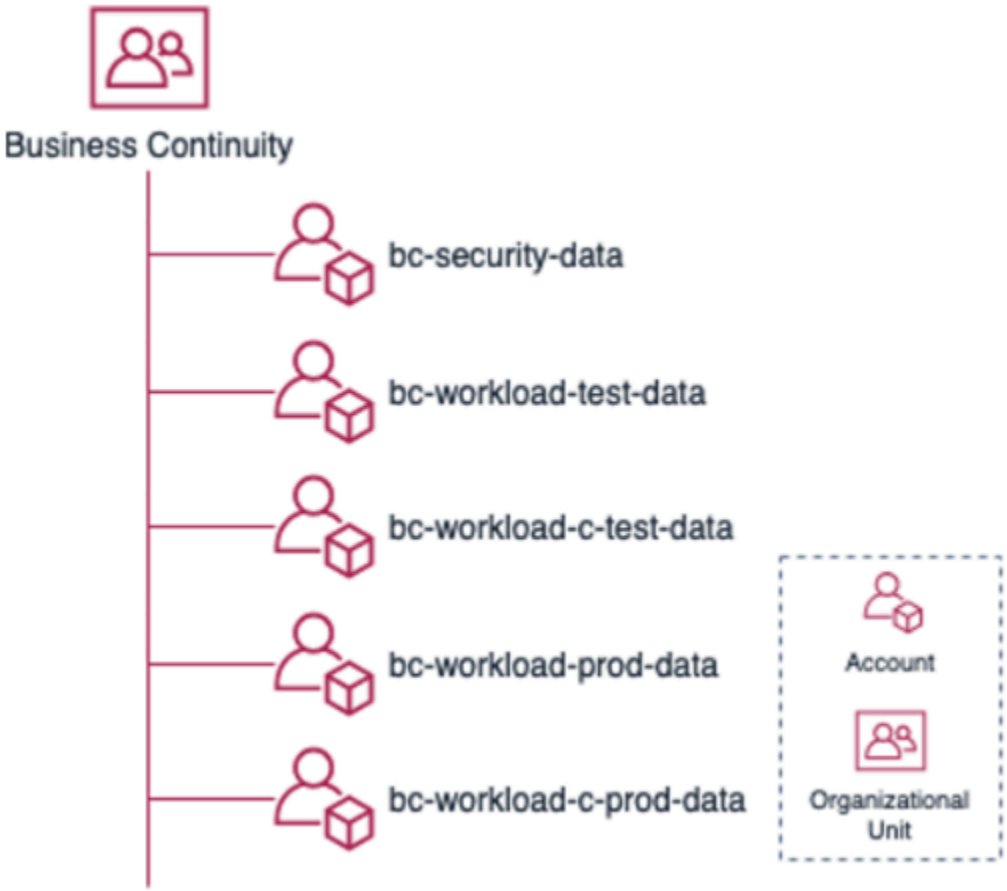
- Place restrictions on the Backup Administrator role so that backup policies for the Business Continuity OU are not altered.
- Implement monitoring notifications to confirm that backups have not been interrupted. Refer to the documentation on AWS Backup monitoring.
- Require all Backup Vaults use AWS Backup Vault Lock in Compliance mode with a minimum retention of 14 days or more.
- Audit backups regularly to ensure compliance of your backup policies. Refer to the documentation on AWS Backup Audit Manager.

Example structures

In this example, a company, Rainbows, has a starter AWS environment that follows the production starter organization guidance. Rainbows has three workloads called A, B, and C. They have separate testing and production accounts for each. The data for workloads A and B are non-sensitive and unregulated. However, the data for workload C is highly sensitive and regulated and must be isolated from other workload data.

Following is an example of the Business Continuity OU for the Rainbows company. Because the data for workloads A and B is non-sensitive and unregulated, you can keep it in the same business continuity (bc) account, `bc-workload-test-data` and `bc-workload-prod-data`. However, for workload C, the business continuity data is isolated in separate accounts, `bc-workload-c-test-data` and `bc-workload-c-prod-data`, because the workload is highly sensitive and regulated.

> **Note**
>
> Some companies choose to keep each workload's data separated in individual accounts for an enhanced security posture, regardless of a regulatory or compliance need.



Business Continuity OU example structure

Organizing workload-oriented OUs

The recommended **Security OU and accounts, Infrastructure OU and accounts, Workloads OU, and Deployments OU** are top-level OUs that contain workloads. This section outlines considerations for organizing these workload-oriented OUs.

Topics

- Workloads
- Workload environments
- Workload accounts
- Production and non-production workload environments
- Workload dependencies across environments
- Production environments accessing non-production
- Non-production environments accessing dependencies

Workloads and environments

This section defines basic terms and concepts related to workloads. Becoming aware of these concepts helps you understand our recommendations for organizing your workload-oriented OUs.

Workloads

Many of your top-level OUs will house collections of applications, cloud resources, and data in the form of workloads. A workload is a discrete collection of components and data that you manage. A workload can be a commercial off-the-shelf (COTS) application or your own custom application and data service.

Composition of a workload

Workload environments

For a given type of workload, you typically have multiple instances. This setup means that you can experiment, develop, and test changes to the workload before you promote and deploy those changes to the production instances of the workload. A given instance of a workload is a *workload environment.*

Whether your workload is a COTS application, a custom application, a custom data service, or a foundational security or infrastructure capability, you often need separate non-production workload environments to support your software development lifecycle (SDLC) processes. You can have multiple SDLC processes depending on the diversity of your workload portfolio and your company organization.

The following example shows multiple environments of a workload across non-production test and production workload environments.

Example of multiple environments of a workload

With COTS applications, you might not perform custom development, apart from implementing custom integrations with your own systems. However, you can experiment with and formally test new versions of the COTS applications in non-production environments before deploying them to production.

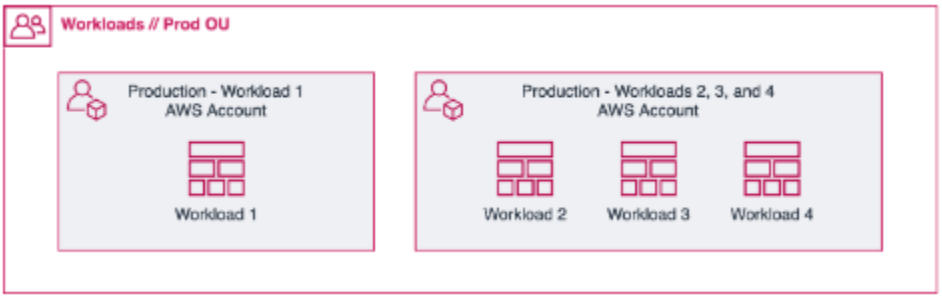
Workload accounts

In your on-premises context, you can refer to the places where your workloads reside as hosting environments. For example, you might have a dedicated production hosting environment in which your production workload environments reside.

In AWS, each of your workload environments is typically contained in an account, where each account is similar to a distinct hosting environment.

Depending on how you choose to scope your workload accounts, you might have a single workload environment per account. Or, you might have multiple workload environments and perhaps multiple workload types in the same workloads account.

The following diagram shows two degrees of scoping production workload accounts. In one example, a workload account is dedicated to a single workload environment. In the other example, multiple workload types reside in a single production workload account.



Example workload accounts with different degrees of scoping

Production and non-production workload environments

We recommend that you isolate production workload environments and data in production accounts housed within production OUs, under your top-level workload-oriented OUs. Apart from production OUs, we recommend that you define one or more non-production OUs that contain accounts and workload environments that are used to develop and test workloads.

Workload dependencies across environments

When you consider the structure of your workload-oriented OUs, you should decide on the extent to which you expect access between production and non-production environments.

Production environments accessing non-production

Generally, workloads deployed to your production environments should not depend on workloads contained in your non-production environments.

Non-production environments accessing dependencies

In non-production environments, it is common for workloads to depend on stable shared application, data, and infrastructure services. Where feasible, we recommend that these shared services be non-production test instances. These non-production test instances should use

test data so that your non-production workloads do not depend on access to your production environments and data.

For example, you can configure workloads in a non-production test environment that depend on integrating with a data service to use a stable, shared test instance of the service that is populated with test data.

However, in some cases non-production environments might need access to production shared services. For example, it's typical for non-production development and test environments to require read-only access to shared source code and artifact management services. Providing access to these shared services enables you to deploy candidate and promoted changes and artifacts to your non-production environments in support of development and testing activities.

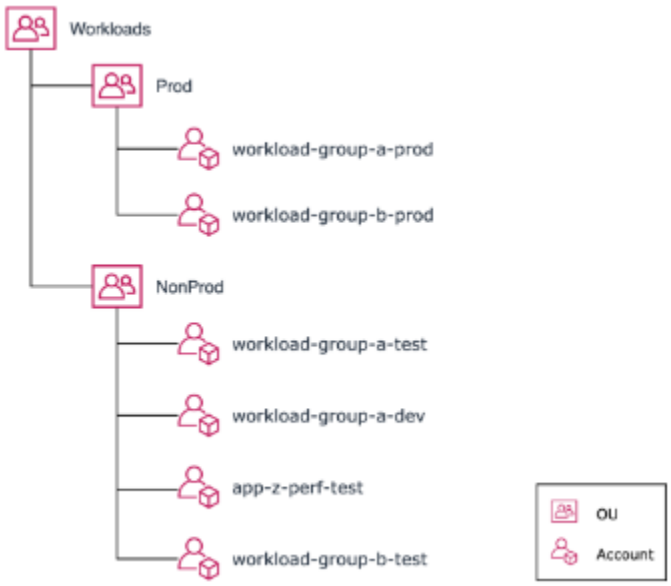
OU structure for non-production environments

You can use OUs to organize your non-production environments in a couple ways.

Option A: Common controls across non-production environments

When non-production workloads require the same set of overall access policies or benefit from being operationally managed together, you can define a single NonProd OU to contain all the accounts that support non-production forms of your workloads.

The following example shows the Workloads OU where a Prod child OU contains production accounts and workloads, and a NonProd child OU combines both development and test accounts and workloads.



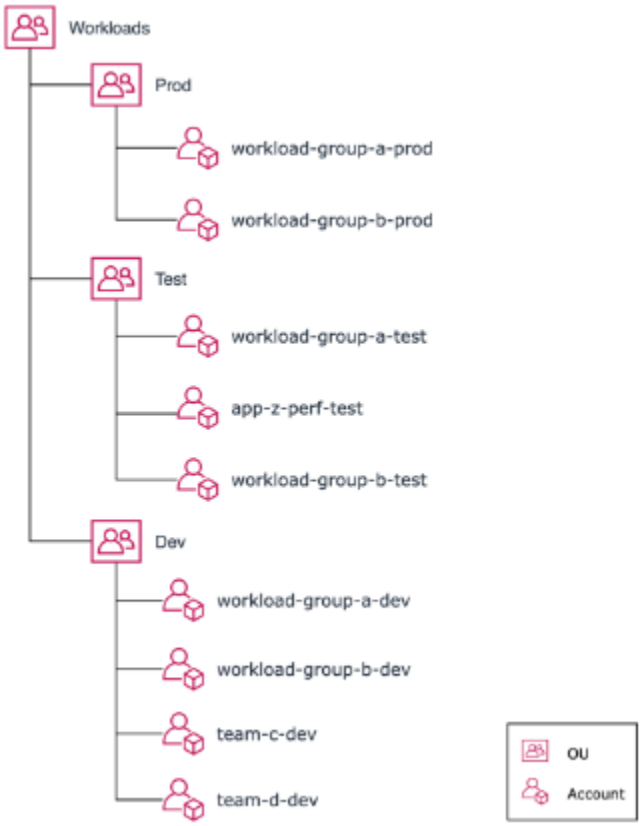
Example Workloads OU with common policies across a NonProd child OU

Option B: Different controls across non-production environments

Sometimes your process for developing and testing changes involves workload environments that have fundamentally different access policies or ways in which you manage and apply foundational resources. In these cases, it makes sense to create distinct OUs to support these diverse requirements.

For example, you want to support development environments that provide teams with more freedom to experiment, iterate, and develop largely on their own (rather than more formally managed and controlled production-like test environments). In this case, overall access policies and management of baseline resources for the development environments is significantly different than those used to support test environments. It makes sense for you to create a distinct OU for development work and another OU for your test workloads.

The following example represents a simple form of this structure where Test and Dev OUs reside adjacent to the recommended Prod OU.



Example Workloads OU with different policies for Test and Dev child OUs

The preceding example shows two different approaches to scoping development environment accounts. One approach is where development environments are aligned with the same groupings of workloads as used in test and production OUs. The other approach is one in which development environments are aligned based on teams.

Extended workload-oriented OU structure

An extended form of the workload-oriented OU structure can be used to support cases in which you need to either organize workloads for visibility and management purposes or apply different security and operational policies to either a workload or group of related workloads.

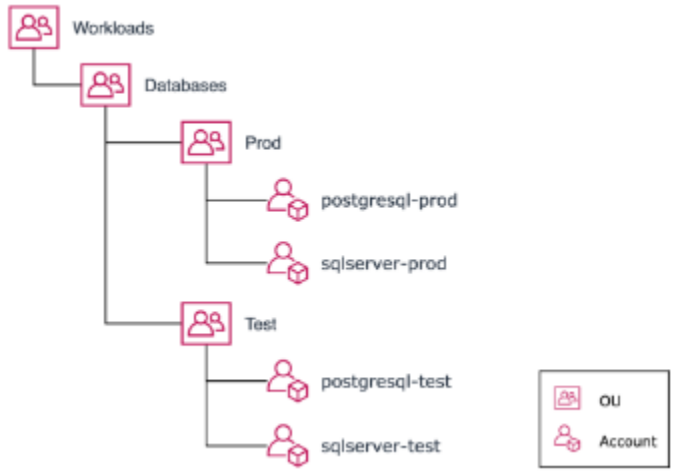
When workloads have diverse security and operational policy requirements, you cannot effectively manage controls and other controls at the level of the workload-oriented OU. By adding child OUs to a workload-oriented OU, you can group related workloads in the same child OU. You can then apply distinct security and operational policies to the child OUs.

For example, a workload or a group of related workloads might benefit from having a distinct allow list of AWS services that is implemented via a service control policy (SCP). This policy might be different than the requirements associated with other workloads. Rather than applying the SCP to each of the related workload accounts, it is recommended that you apply the SCP to an OU that groups the related accounts.

Grouping related workloads

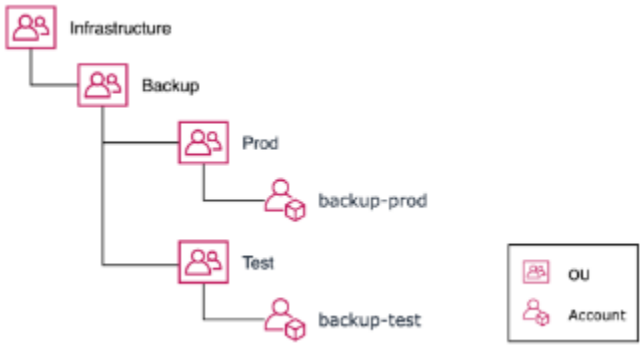
When you have groups of related workloads that require the same overall set of security and operational policies, you can create a child OU for each group of workloads.

For example, if you manage a series of database services that are shared across your organizations and have common security and operational policy requirements, you might find value in grouping those data services under a common child OU.



Group of workloads with distinct policy requirements

The following example represents a shared backup capability you can provide across your AWS environment. If this capability requires a set of security and operational policies that are distinct from other infrastructure workloads, then you can allocate a distinct OU for this workload.

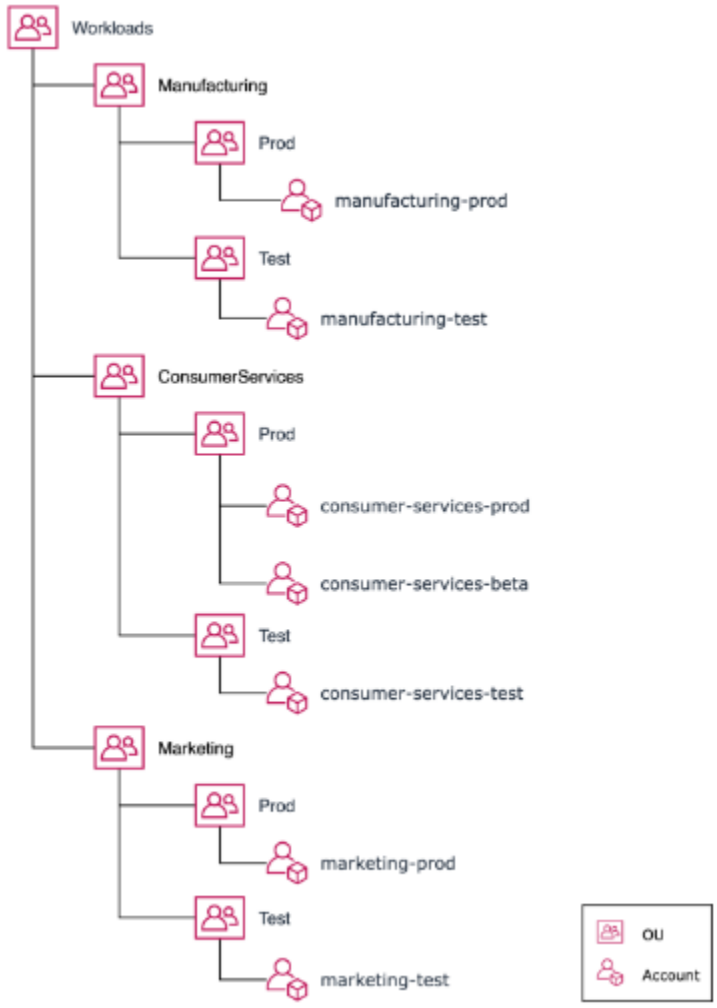


Single workload with distinct policy requirements

Separating business units with significantly different policies

If you have largely autonomous business units (BUs) that manage workloads in your common AWS organization and the BUs have significantly different security and operational policies, you can create a child OU under your Workloads OU for each BU.

In the following example, each BU is provided with its own OU so that different SCPs and/or operational policies can be applied independently from the other OUs.



Example business unit separation

How does AWS Control Tower establish your multi-account environment?

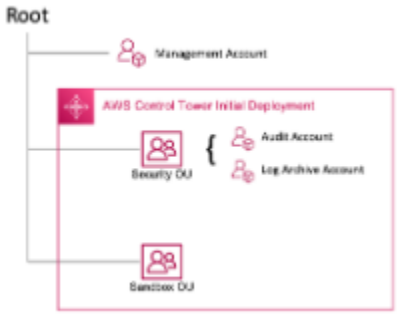
AWS Control Tower offers a straightforward way to set up and govern an AWS multi-account environment, following prescriptive best practices. AWS Control Tower *orchestrates* the capabilities of several other AWS services, including AWS Organizations, Service Catalog, and AWS IAM Identity Center. This section describes at a high level how AWS Control Tower establish a multi-account environment and landing zone. Your landing zone is a well-architected multi-account environment for all of your AWS resources. You can use this environment to enforce compliance regulations on all of your AWS accounts.

Topics

Establish your multi-account environment with AWS Control Tower

When you set up your multi-account environment using AWS Control Tower, it creates two OUs.

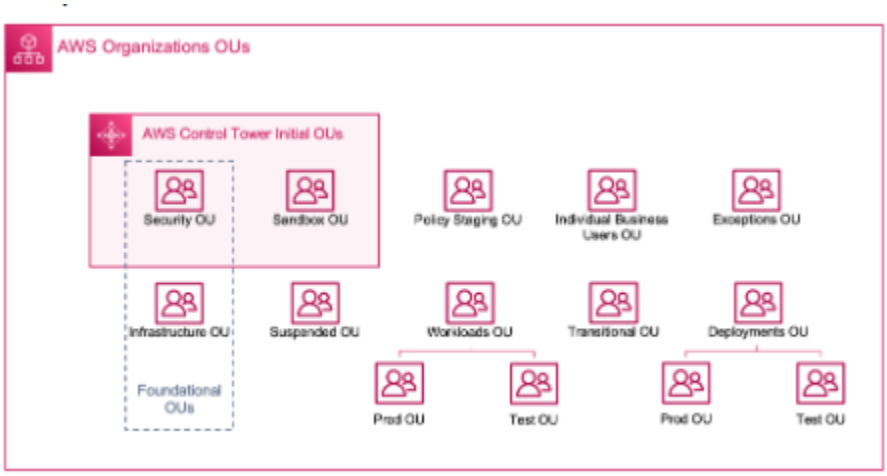
- Security OU—Within this OU, AWS Control Tower creates two accounts:
  - Log Archive
  - Audit (This account corresponds to the Security Tooling account discussed previously in the guidance.)
- Sandbox OU—This OU is the default destination for accounts created within AWS Control Tower. It contains accounts in which your builders can explore and experiment with AWS services, and other tools and services, subject to your team's acceptable use policies.



OUs and accounts created by AWS Control Tower

AWS Control Tower allows you to create, register, and manage additional OUs to expand the initial environment to implement the guidance.

The following diagram shows the OUs initially deployed by AWS Control Tower. You can expand your AWS environment to implement any of the recommended OUs included in the diagram, to meet your requirements.



OUs initially deployed by AWS Control Tower

Next steps for setting up your multi-account environment

To get started with AWS Control Tower, see Getting started with AWS Control Tower. We recommend that you review the prerequisites and next steps required to establish your multi-account environment on AWS.

For complete guidance on establishing your multi-account environment, review the guidance included in this whitepaper.

Implementation

Topics

Getting started with your multi-account environment

The earlier sections of this guide covered the benefits of using multiple accounts to organize your AWS environment, this section dives deeper into some implementation options you can use to build your multi-account environment.

Your environment might start with a small number of accounts, and as the number of accounts in your environment increase, you will need automations to manage your cloud environment. These services offer features to help you manage your multi-account environment at scale.

New customers

Managing your own operations

When starting on AWS, you should evaluate your business goals and determine your future operating model, and decide on which technology to use when building your multi- account environment. If you are planning on managing your own environment, start by evaluating AWS Control Tower.

AWS Control Tower is a service built based on the guidance included in this paper, making it the preferred solution for new-to-the-cloud customers looking for a service-managed environment. AWS Control Tower offers a simplified experience and automatically deploys your initial environment, helping you manage the multi-account environment efficiently, using other AWS services. For a complete list of these services, refer to the AWS Control Tower user guide.

Operations managed by AWS or AWS Partners

AWS and AWS Partners can help you operate, update, monitor, or deploy your environment and your workload infrastructure, with AWS Managed Services and AWS Partners have offerings that can help with your operations.

Hybrid operations (mix of customer and AWS managed)

If your portfolio includes different operating models, you will need to consider a combination of the approaches described above. Proving the flexibility to decide what type of operations you will manage on your environment, and which operations you would like to be overseen. For example, you could manage the account vending process and the environment set up using AWS Control Tower, and use Amazon Managed Services for managing the operations related to moving to the cloud.

Existing customers

If you are not operating your current environment using AWS Organizations, or you are managing an AWS account structure that differs significantly from the strategies laid out in this whitepaper, consider examining your operational model and re-organizing your accounts, and consider operational and organizational changes to align with these strategies.

AWS has seen many challenges for customers because they have grown organically without employing these recommendations, or by implementing before these recommendations were available. These challenges include the following:

- Operational inefficiencies and difficulties, such as unintentionally encountering account quotas when operating many workloads in a small number of accounts due to lack of visibility of the shared quotas
- Use of service roles by humans, and exhaustion in the number of users due to the number of these roles needed to operate many workloads in a small number of accounts
- Unnecessary complexity due to the large number of accounts managed to operate one workload. This complexity can be part of root causes in incidents within security, availability, and other areas
- Lack of automation in the provisioning of accounts, causing large delays in implementing new workloads

- Challenges in managing the email contacts required for notifications about each account (due to lack of automation of account ownership attribution to the workload operators)
- Challenges in ensuring data governance and residency of data where customers have chosen to use a central account and region for storage, and use of this data in an effort to minimize the number of accounts they are operating

Embrace infrastructure as code

One of the foundational operational capabilities that will enable your ability to manage change effectively is to use infrastructure as code technologies and automation to build and deploy your workloads in AWS. This basic capability might be perceived as being part of your Infrastructure OU, but these are foundational capabilities for all OUs.

Examine your operational model

Amazon has "commitment to operational excellence" as one of its four guiding principles of who we are. We have an operational model that enables rapid decisions and autonomy. The way you operate your business is a primary consideration in how you organize your environment. In many companies, this doesn't change regularly, and changes should always be made with careful consideration of the business impact of the change. Your Operating Model could be aligned by business unit, regulatory restrictions, area of expertise, and/or organic growth. You might have a need for multiple operational models, for different businesses or business units. The ability for you to meet your goals using AWS is fundamentally affected by your ability to run your operational model.

Implement identity management and access controls and other security capabilities

The Security OU is a foundational OU. Using a central identity provider enables you to centrally manage both the identities of the people and services that will need to access resources in your environment, and the permissions associated with that access. To prevent a single source of compromise, consider having more than one identity provider. If you do not currently have an identity provider, consider AWS IAM Identity Center. This also provides a central place to enforce multi-factor authentication and provide federated access into your environment. Evaluate your capabilities and operational model between your identity management and Amazon S3 permissions management (also known as access controls).

Next, set up the security capabilities (for example: , encryption and key management, secrets management, and incident response capabilities) in the appropriate environment. You can then use these operational capabilities as you reorganize other parts of your operations. Decide if you want to have a security "reacentralized log storaged only" account to manage and use cross-account access in your environment, or use federated roles directly to access individual accounts in your environment.

The following capabilities can be built as you scale and deploy workloads, or in preparation for their use as you move workloads:

- The additional security capabilities of vulnerability and threat management (as part of security tooling)
- Data de-identification and data isolation (which is normally part of the individual workloads in your Workloads OU.)
- Patching (which might use your Deployments OU to patch golden machine and container images)
- Forensics

Separate the production workload environment from non-production environment(s)

Next, consider implementing hard isolation between your production environment and your non-production environments by implementing them as separate accounts. This reduces your risk of exposing production data in non-production environments, but might increase the complexity of how you perform deployments. As a result, this change could affect many of your operational capabilities, including:

- Network connectivity, in your Infrastructure OU and accounts, and Workloads OU.
- Network security, in your Workloads and Security OUs
- Application security, in your Workloads, Deployments, and Security OUs
- Tagging, in all OUs
- Service onboarding and cloud financial management, in all OUs
- Rollout/rollback and change management capabilities, in your Deployments and Workloads OUs
- Developer experience and tools, in your Sandbox OU, as well as in your Deployments and Workload OUs

It is not uncommon to have basic connectivity to and from your local networks and the internet shared between your production and non-production workloads. If you have deployment tooling that is tightly integrated with your current environment, you can integrate the existing implementation as you move the workload, or you can implement the Deployments OU to also deploy to your existing environment while you move them to your new environment.

Networking considerations in a multi-account environment

The number of VPCs a customer operates is usually related to the number of accounts, regulatory requirements (such as the payment card industry (PCI), and compliant/non-PCI compliant), and staged environments (such as prod, dev, and test). With an increasing number of VPCs (account isolated or not), give careful consideration to cross-VPC connectivity management. This is an essential part of the customer's network operation. Additionally, IP address management becomes a key contributing factor to enabling scalability and future growth. You might consider designs that are built around IPv6 adoption that can be driven by the need to scale your network, or by a strategic initiative. Current recommendations for three specific areas in cross-VPC and hybrid connectivity can be grouped by:

- **Network connectivity** — Interconnecting VPCs, on-premises networks at scale, and choosing the right tool for the use case. For intra-AWS connectivity, VPC peering, AWS Transit Gateway, and AWS PrivateLink are just a few options which help with different use cases.
- **Network security** — Building centralized or distributed inspection points for accessing the internet and VPC-to-VPC traffic needs to account for the different options, such as AWS Network Firewall, AWS Gateway Load Balancer, and AWS Web Application Firewall.
- **DNS management** — Resolving DNS within the AWS and hybrid environments at scale involves both right-sizing and choosing the deployment model that best suits the organization's scaling and growth goals. Inbound and outbound Route 53 Resolver endpoints can be centralized or distributed, depending on the operations and management models, and involve different needs across the AWS network environment.

Separate the workload environments to align with the operational organization units

To achieve agility and autonomy, your environment will separate into organizational boundaries. Align your workload environments (both production and non-production) with these organizational boundaries to ensure you enable further agility and autonomy. As you grow and expand, it is

common to create new boundaries and move workloads to ensure you continue to have these business capabilities. This could further affect your operational capabilities, such as backups and disaster recovery, support, template management, records management, and sorting and searching via metadata in your workloads OU. Consider centralizing the management of these capabilities to simplify the realignment. You can use tag policies as a means to classify the workloads. You can use tag policies as a means to classify the workloads. You can use AWS Backup to back up to a designated location in your environment and allow you to centralize protection and monitoring of your backups.

Create the additional organizational units to enable other capabilities

You should consider creating an Exceptions OU. Also, consider creating a Policy Staging OU. As your business might either acquire or divest parts of your business, consider having a Transitional OU to enable an area to change the policies to align with new requirements. As you deprecate accounts, you will want a Suspended OU to contain these accounts until you are comfortable having them permanently removed. You should consider creating a Individual Business Users OU to enable business users and teams who need access to manage AWS resources directly, rather than management by the Workload OU.

Other considerations for implementing these changes

The reorganization will require movement of accounts, or migrations of workloads between accounts if you have deployed workloads in accounts that don't align with your desired operational model. There are mechanisms to move accounts between organizations and organizational units, but if your new approach indicates some workloads in an account should belong to one organizational unit and other workloads belong to another organizational unit, you will have misalignment. To align the accounts with the operational model, you will have to migrate workloads between existing accounts, or to new accounts. The methods and capabilities to migrate between accounts are similar to accomplish these migrations.

Available services

AWS Organizations

AWS Organizations provides the underlying infrastructure and capabilities for customers to build and manage their multi-account environments. Using AWS Organizations, customers can automate AWS account creation and management; govern access within the organization to AWS services,

resources, and Region using preventative controls; centrally manage policies across multiple AWS accounts (SCPs, Tag Policies, AWS Backup, ML opt-out); configure multi-account capabilities for AWS services (such as AWS Config, AWS CloudTrail, AWS CloudFormation, GuardDuty, Amazon Macie, and IAM Identity Center); share resources across accounts; and consolidate their bill.

AWS has the following resources available for help you establish your multi-account environment using AWS Organizations:

- AWS Organizations features
- Organizations supported multi-account services
- Organization Quotas

AWS Control Tower

AWS Control Tower provides a simplified way to set up and govern a secure, multi-account AWS environment based on the guidance in this paper. AWS Control Tower automates the creation of your multi-account environment using AWS Organizations, instantiating a set of initial accounts and with some default controls and configurations for the environment. Although AWS Control Tower reduces flexibility, it also provides automations to manage your cloud environment efficiently.

> **Note**
>
> The OU structure that AWS Control Tower initially deploys is slightly different from the guidance in this paper, review AWS multi-account environment with Control Tower for a detailed implementation and mapping between the AWS Control Tower implementation and the guidance offered in this paper.

AWS has the following resources available for help you establish your multi-account environment using AWS Control Tower:

- Getting started with AWS Control Tower
- AWS Control Tower Quotas

AWS Managed Services

AWS Managed Services (AMS) uses AWS services and a growing library of automations, configurations, and run books, to provide an end-to-end operational solution for both new and existing AWS environments. AMS covers the people element of operating the technology that AWS services provide.

For additional information, refer to AWS Managed Services features.

Conclusion

If you are in the early stages of adopting AWS, you can use these best practices to start implementing an AWS environment structure that is sufficient to meet your initial needs. As your adoption of AWS expands and your requirements increase, you can be confident that your AWS environment can be expanded to meet those needs without requiring significant restructuring. If you already have an AWS environment in place, you can use these best practices to assess its current state. By doing so, you can determine if you're fully realizing the benefits of using multiple OUs and AWS accounts and, if necessary, you can make plans to enhance your current environment.

In either case, your AWS sales team and AWS Partner Network (APN) partners are ready to help you apply these best practices to meet your business needs.

Contributors

Contributors to this document include:

- George Rolston, Sr. Solutions Architect, Amazon Web Services
- Todd Gruet, Sr. Solutions Architect, Amazon Web Services
- Matheus Arrais, Sr. Solutions Architect, Amazon Web Services
- Paul Bayer, Principal Consultant, Amazon Web Services
- Sam Elmalak, Principal Solutions Architect, Amazon Web Services
- Ilya Epshteyn, Director, Identity Solutions, Amazon Web Services
- Christopher Kampmeier, Senior Solutions Architect, Amazon Web Services
- Tomas Riha, Senior Solutions Architect, Amazon Web Services
- Dave Walker, Principal Solutions Architect, Security and Compliance, Amazon Web Services
- Alex Torres, Sr. Solutions Architect, Amazon Web Services
- Rodney Lester, Principal Solutions Architect, Amazon Web Services
- Brandon Wu, Senior Security Solutions Architect, Amazon Web Services
- Nathan Case, Security Strategist, Amazon Web Services
- Brian Mycroft, Enterprise Technologist, Amazon Web Services
- Jason DiDomenico, Sr Solutions Architect, Amazon Web Services
- Emeka Enekwizu, Sr. Solutions Architect, Amazon Web Services
- Brian Hesseling, Sr. Solutions Architect, Amazon Web Services

Additional support

For additional information, see:

- Cloud Operations on AWS
- Security, Identity, and Compliance on AWS
- AWS Best Practices for Security, Identity, and Compliance
- AWS Well-Architected
- AWS IAM Permissions Controls
- Establishing your Cloud Foundation on AWS
- AWS Control Tower Workshop
- AWS Solutions-Focused Immersion Days

Document history

To be notified about updates to this whitepaper, subscribe to the RSS feed.

Change	Description	Date
Updated	Updated guidance to reflect re:Invent 2024 launches.	April 30, 2025
Updated	Updated for latest recommendations and best practices.	January 14, 2025
Updated	Updated for technical accuracy.	March 28, 2024
Updated	Updated break-glass guidance and included disaster recovery guidance.	March 15, 2023
Updated	Updates to include the latest best practices for managing a multi-account environment.	July 26, 2022
Updated	Updated guidance for existing customers getting started with their multi-account environment.	March 31, 2022
Updated	Updated guidance for establishing a multi-account environment.	July 19, 2021
Initial release	Whitepaper first published.	March 18, 2021

Notices

Customers are responsible for making their own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice, and (c) does not create any commitments or assurances from AWS and its affiliates, suppliers or licensors. AWS products or services are provided "as is" without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.