OU structure in regulated AWS landing zones: an example from the pharmaceutical industry

AWS Prescriptive Guidance

AWS Prescriptive Guidance: OU structure in regulated AWS landing zones: an example from the pharmaceutical industry

Table of Contents

OU structure in regulated AWS landing zones: an example from the pharmaceutical industry

Katja Mueller, Petar Forai, and Keval Sheth, Amazon Web Services

AWS provides several Landing Zone Accelerator configurations that support specific industries, including healthcare. The Landing Zone Accelerator for Healthcare is used in conjunction with AWS Control Tower to ease the management and governance of a multi-account environment that is aligned with AWS best practices and multiple global compliance frameworks. One important aspect of orchestrating governance is to group AWS accounts together by using organizational units (OUs), so you can administer them as a single unit.

This guide discusses best practices and considerations for redesigning existing OU structures as your company's cloud maturity grows. It introduces a practical example based on the AWS Control Tower implementation for a large pharmaceutical company, and discusses the lessons learned from that implementation. AWS landing zone design is not a one-time effort. It can and should be allowed to evolve. This evolution can be achieved successfully and efficiently by applying the AWS best practices and tips highlighted in this guide.

Overview

We worked with a multinational pharmaceutical company to run proof-of-concept (PoC) activities on AWS to explore how they could migrate their applications from on premises to the AWS Cloud. As they started scaling and accelerating their migration workflow to include production workloads, it became clear that they needed a regulated and compliant AWS landing zone to support their goal. We used AWS Control Tower to design and implement a new AWS landing zone.

An important aspect of a new AWS landing zone and its organization is the structure of its *organizational units* (OUs). An OU is a logical grouping of AWS accounts created by using AWS Organizations. You can use OUs to organize AWS accounts into a hierarchy and apply management and governance controls consistently and more easily.

You can attach policy-based controls to an OU and to AWS accounts. Child OUs within an OU automatically inherit those controls. Therefore, OUs play a critical role in managing security and governance in AWS Organizations.

A *policy* is a JSON document that includes one or more statements that define the controls you want to apply to a group of AWS accounts. AWS Organizations currently supports four types of policies, also known as *service control policies* (SCPs). An SCP defines the AWS services and actions that are available for use in different AWS accounts. For example, you can use an SCP to require Amazon Elastic Compute Cloud (Amazon EC2) instance launches to use a specific instance type.

Policy types

SCP policy types include the following:

- Allow lists and deny lists are complementary strategies that you can use to apply SCPs to filter the permissions that are available to accounts.
- Tag policies help you maintain consistent tags, including the preferred case treatment of tag keys and tag values across accounts.
- Backup policies configure backups for supported resource types, such as the time window of the backup and destination vaults for backups across AWS Regions.
- AI services opt-out policies enable or disable the continuous improvement of AWS artificial intelligence (AI) services globally.

Policy inheritance behavior: When you attach a policy to a specific OU, the accounts that are directly under that OU or any child OU inherit the policy. When you attach a policy to a specific account, the policy affects only that account. You can overwrite inherited policies by introducing exception policies. Inheritance explicitly allows all permissions to flow down from the root (OU) to every AWS account in that OU and child OU, unless you explicitly deny a permission. To deny a permission, you create an additional policy and attach it to the appropriate OU or AWS account. For more information about policy inheritance and exceptions, see the AWS Organizations documentation.

AWS Control Tower also provides its own set of detective and preventive controls (also called *guardrails*) by using the OU structure. AWS Control Tower preventive controls prevent actions by using the SCPs in AWS Organizations. Detective controls report on configurational drift against the control by using AWS Config. For more information about these controls, see the AWS Control Tower documentation.

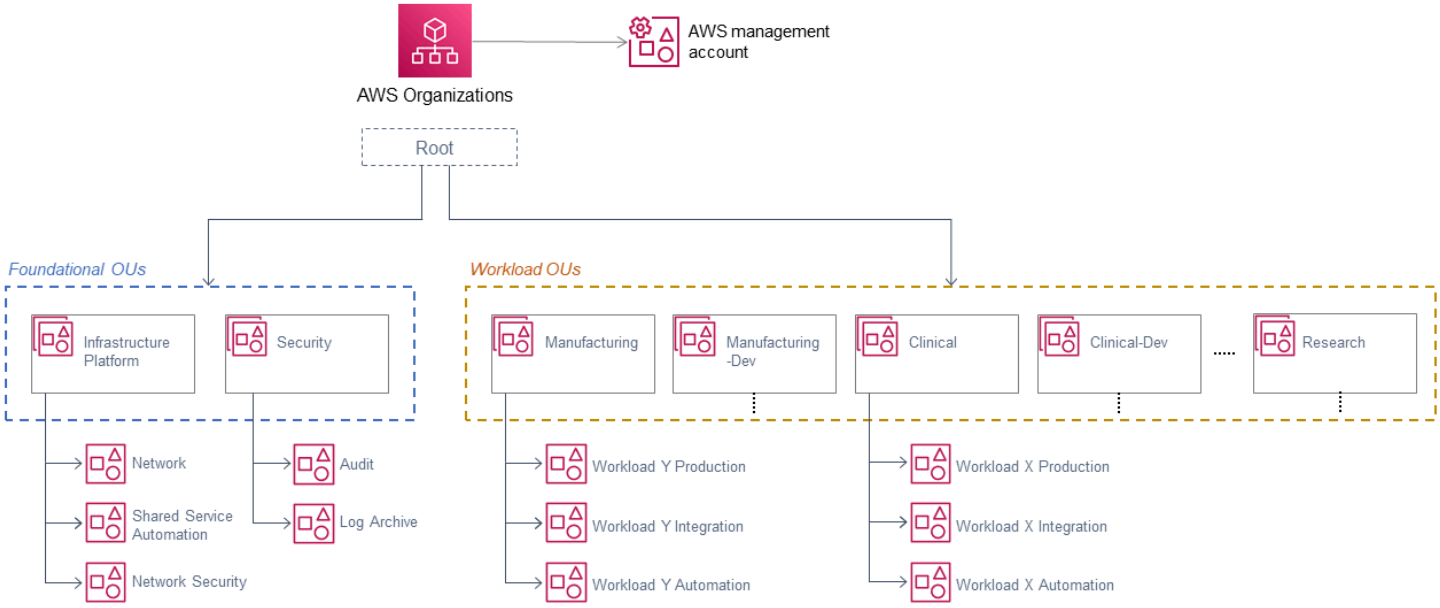
You can also enable AWS Security Hub to automate security best practice checks, aggregate security alerts into a single place and format, and understand overall security posture across all your AWS accounts.

OU design: phase 1

For the multinational pharmaceutical company in our example, the initial design of the organizations and OUs in AWS Organizations closely followed AWS recommendations for setting up AWS Control Tower. For an example, see the Landing Zone Accelerator on AWS for Healthcare. AWS Control Tower initially provisioned a simple OU structure with common foundational OUs, as described in the blog post Best Practices for Organizational Units with AWS Organizations, including the Security OU, the Platform Infrastructure OU, and company-specific OUs.

Architecture design

The following diagram shows the initial OU architecture.



Security OU

The Security OU broadly groups AWS accounts related to security functionality together and uses two accounts (Audit and Log Archive) to store security operational data for central logging and auditing access to the environment. AWS core security services such as Amazon GuardDuty and AWS Security Hub reside in the Audit account.

Infrastructure Platform OU

The Infrastructure Platform OU groups together AWS accounts that provide the infrastructure foundation. Initially deployed within this OU are the AWS accounts for the central networking components (gateways, firewalls, central networking hub, and similar services).

Additional OUs

Other, company-specific OUs (such as a Clinical OU) augment the foundational OUs within a low-level hierarchy. Workloads are implemented with a multi-account structure and with separate environments within those OUs.

Several considerations drove this initial design:

- Nested OUs were not available at that time in AWS Control Tower and required extensive customization.
- Initial workloads designated for the cloud focused on particular aspects of the company such as clinical trials or manufacturing equipment analytics (functional views).
- The company differentiates between five workload environments (development, validation, integration, training, and production). The company needed a playground for developing applications without the strict governance by AWS controls that production workloads required. Development OUs such as the Manufacturing-Dev OU were assigned for this purpose.
- Workload automation was part of each application's ecosystem and did not need separation.
- Infrastructure qualification (IQ) and GxP compliance processes did not require a distinction of AWS controls at the OU level.

OU design: phase 2

The pharmaceutical company in our example accelerated into a new phase of cloud maturity by deploying qualified production workloads into the existing OUs. This initiated a review of the initial design, and the phase 1 structure was challenged as more workloads migrated to the regulated AWS landing zone.

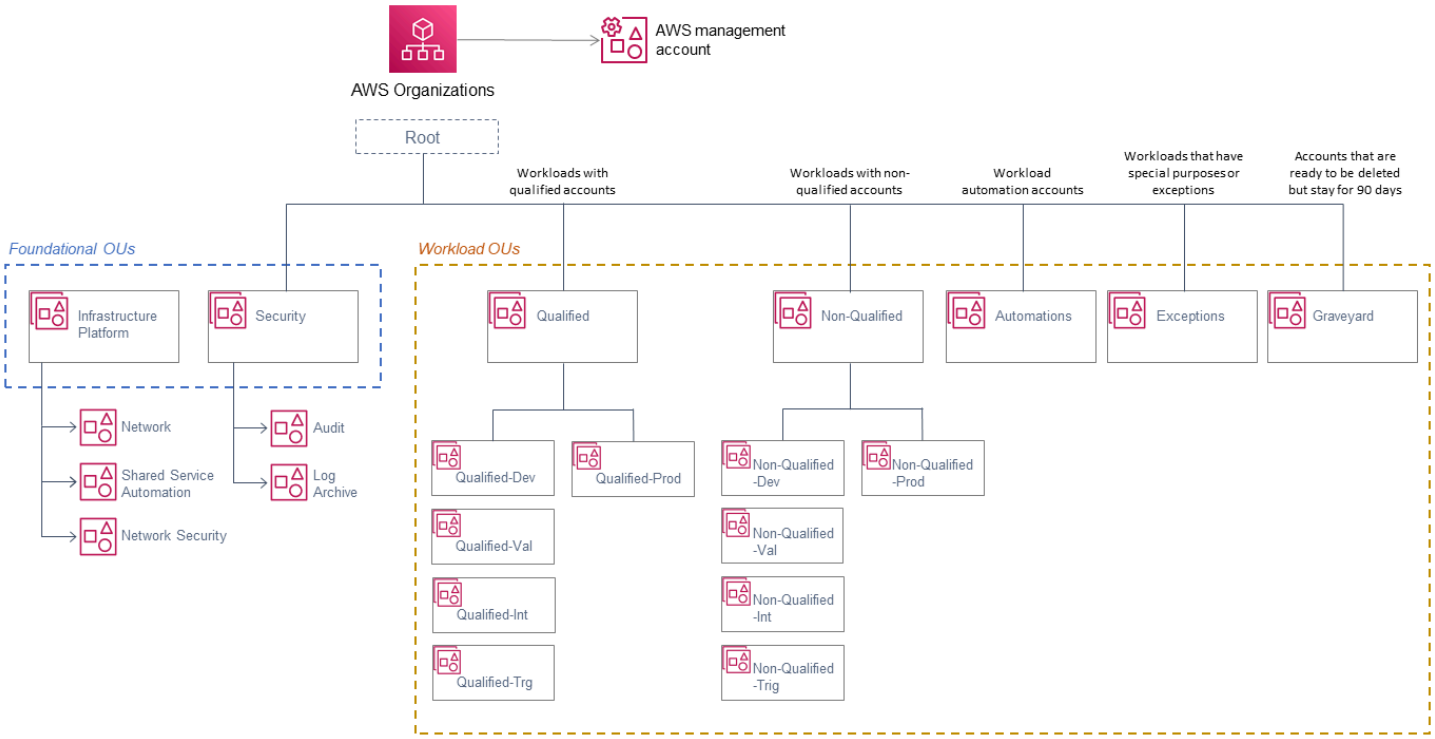
The following new requirements and insights became important:

- The company implemented data sharing model workloads, so applications acquired a multi-purpose nature that could no longer be assigned to separate OUs such as Clinical or Manufacturing.
- Qualification (in particular, ongoing qualification) became a vital aspect of many workloads. These workloads had to be integrated into operational processes so they could follow security best practices more easily. Qualified workloads required more stringent AWS controls, which were set at the OU level in phase 1.
- Nested OU functionality became available in AWS Control Tower.
- Upskilling and experience resulted in a better understanding of which specific policies were relevant for workloads.
- The company defined and agreed on an operational model that was based on responsibility alignment.
- Workload segmentation and structuring blueprints matured and were adopted for workload migrations.

As a result of these, a new design was implemented in phase 2 and AWS accounts migrated to that new structure. This new structure includes the OUs described in the following sections.

Architecture design

The following diagram shows the OU architecture for phase 2.



Security OU

The Security OU contains AWS accounts related broadly to security functionality and uses two accounts (Audit and Log Archive) to store security operational data for central logging and auditing access to the environment. AWS core security services such as Amazon GuardDuty and AWS Security Hub reside in the audit account. This OU remains unchanged from the original design.

Infrastructure Platform OU

The Infrastructure Platform OU contains foundational infrastructure accounts such as networking and shared automation across the AWS landing zone. This OU remains unchanged from the original design.

Qualified OU

The Qualified OU contains workloads that require a qualified infrastructure such as stringent change management, qualification, and validation.

Non-qualified OU

The Non-Qualified OU contains workloads that don't have GxP requirements or aren't business-critical.

Automations OU

The Automations OU contains shared resources for workload automation, such as continuous integration and continuous delivery (CI/CD) pipelines for infrastructure management. Depending on requirements, automation can either be split across environments or hosted in a single AWS account.

Exceptions OU

The Exceptions OU contains workloads that require special treatment that would otherwise be prevented by policies. For example, widely accessible and readable Amazon Simple Storage Service (Amazon S3) buckets would belong in the Exceptions OU.

Graveyard OU

The Graveyard OU contains AWS accounts for workloads that will be deleted. Policies in these accounts should be removed for effective and simple administrative access until the account expires or is deleted.

OU migration and verification

For the multinational pharmaceutical company in our example, cloud platform engineering and change management teams agreed on a migration plan and schedule. AWS accounts were reviewed and ranked according to business impact and criticality as part of the planning.

The new OU structure was deployed side by side with the existing OU structure to allow for staging of new policies and incremental migration. We created new empty OUs with child OUs and assigned AWS Organizations policies and AWS Control Tower controls to those child OUs.

The AWS Organizations policies were migrated manually by applying the desired policies on the new structure. We used manual spot checking to verify AWS Organizations policies and automated mechanisms to verify AWS Control Tower controls. AWS accounts were moved to the new OUs only after these checks were successful.

Account migration was semi-automated and followed a staggered or batched approach. The initial migration targeted AWS accounts that were considered less critical, using a batch of five AWS accounts per migration run. Migration batches did not exceed 10 accounts. When the accounts were migrated, reviewers and testers used the AWS Control Tower console to verify that these accounts were moved to the correct OU and monitored AWS CloudTrail logs for errors. Reviewers also checked the AWS Service Catalog and AWS Control Tower consoles for any drift or accounts in tainted or unknown states.

Changes to OU placement of accounts did not affect the connectivity or accessibility of resources. No outage in a production application was reported.

Lessons learned and best practices

- **OU structure design is not a one-time effort.** As a company increases cloud adoption and migrates additional workloads into the AWS landing zone, its OU design (and implicitly its concept of policies) will also naturally evolve.
- **Don't mistake OUs for folders; consider them a target for policies.** The OUs and their hierarchy provide the structuring element for AWS accounts and should always be treated as containers for policies. We recommend that you place all AWS accounts that require the same set of policies into the same OU. This guideline also extends to nested OUs (OUs within OUs).

  AWS environments that require centrally shared functionality and cross-workload data sharing capabilities (which are frequently seen in enterprise data platforms) are easier to accommodate in an OU structure that is not based on lines of business (LOB) function classification. For example, a production environment for a manufacturing application is no different from a production environment for a clinical trial analytics application in terms of policies in AWS Organizations.
- **Respect and use inheritance.** When you attach a policy to a specific OU, AWS accounts that are directly under that OU or under any child OU inherit the policy. When you attach a policy to a specific AWS account, the policy affects only that AWS account.

  The migration effort from one OU structure to another depends on how extensively existing policies were configured at the OU or AWS account level. Another important factor is how much policy inheritance was used in the existing OU hierarchy, or if the inheritance was broken. The complexity of migration increases with the implementation of irregular or deviating inheritance paths. For example, if you apply policies at the individual AWS account level or make frequent policy exceptions (which break inheritance), migrating those policies into a new structure will take significantly more effort. In such cases of high complexity, we recommend that you invest the time to review and redesign policy inheritance during migration planning.
- **Take care of both AWS Organizations policies and AWS Control Tower controls.** The OU structure is shared between AWS Control Tower and AWS Organizations. AWS Control Tower provides its own set of detective and preventive controls. These controls apply at the AWS account or OU level. AWS Organizations orchestrates policies on the OU level. In an OU migration we recommend that you migrate AWS Organizations policies first, because these carry more weight for compliance. We recommend that you apply AWS Control Tower controls to the new OU structure in the second migration step.

- **It takes time to update AWS accounts.** You must re-enroll existing AWS accounts individually into the new OU structure by using AWS Control Tower. This takes time. If you have a large number of accounts, we recommend that you streamline this work through automation to control and automate the OU placement of AWS accounts. Here are two example scenarios:
  - *Manual change for a small migration*: The migration lead reassigns each AWS account from its old OU to its new OU in the AWS Management Console. When that reassignment is complete for all AWS accounts, the migration lead opens AWS Control Tower either for each AWS account separately or for all OUs. Re-enrolling AWS accounts in AWS Control Tower requires 10-15 minutes for each AWS account depending on the number of AWS Regions used within the account. AWS Control Tower allows up to five concurrent operations of this kind to run in parallel.
  - *Custom change automation:* Automation simplifies and saves effort. For example, you can automate the management of an AWS account lifecycle from its creation to migration and termination. You can use AWS Control Tower Account Factory to change the OU assignment for an AWS account and run the re-enrollment process. This automation supports the large-scale migration of hundreds of AWS accounts.

Resources

- AWS Organizations User Guide (AWS documentation)
- AWS Organizations terminology and concepts (AWS documentation)
- Service control policies (SCPs) (AWS documentation)
- AWS Organizations API reference (AWS documentation)
- AWS Control Tower User Guide (AWS documentation)
- Introducing Landing Zone Accelerator for Healthcare (AWS blog post)
- Managing the multi-account environment using AWS Organizations and AWS Control Tower (AWS blog post)
- Best Practices for Organizational Units with AWS Organizations (AWS blog post)
- AWS Security Reference Architecture (AWS SRA) (AWS Prescriptive Guidance)
- Establishing Your Cloud Foundation on AWS (AWS whitepaper)
- Organizing Your AWS Environment Using Multiple Accounts (AWS whitepaper)

Document history

The following table describes significant changes to this guide.

Change	Description	Date
Initial publication	—	March 28, 2023