AWS Control Tower
User Guide
## Table of Contents

What Is AWS Control Tower? ................................................................. 1  
Features .......................................................................................... 1  
Related Services ............................................................................ 1  
Pricing ............................................................................................ 1  
Are You a First-Time User of AWS Control Tower? ......................... 1  
How It Works .................................................................................. 2  
  Structure of an AWS Control Tower Landing Zone ...................... 2  
  What Happens When You Set Up a Landing Zone ....................... 2  
  What Are the Shared Accounts? .................................................... 3  
How Guardrails Work .................................................................... 8  
How AWS Regions Work With AWS Control Tower ....................... 9  
How AWS Control Tower Works With Roles to Create and Manage Accounts ............................................ 9  
How AWS Control Tower Works With StackSets ............................ 10  
AWS Control Tower and VPCs ....................................................... 10  
  CIDR and Peering for VPC and AWS Control Tower ................. 11  
  Options for VPC Peering in AWS Control Tower .................... 12  
  Notes on VPC and CIDR ............................................................ 12  
Plan Your Landing Zone ............................................................... 13  
  Launching AWS Control Tower in an Existing Organization ....... 13  
  Summary of Capabilities ............................................................. 13  
  Launching AWS Control Tower in a New Organization ............ 14  
Setting Up ...................................................................................... 15  
  Sign up for AWS ........................................................................... 15  
  Create an IAM User ...................................................................... 15  
  Set up MFA .................................................................................. 17  
Next Step ...................................................................................... 17  
Getting Started ............................................................................. 18  
  Prerequisite: Automated Pre-Launch Checks for Your Master Account .................................................... 18  
  Step One: Create Your Shared Account Email Addresses ........... 18  
  Step Two: Set Up Your Landing Zone .......................................... 19  
  Next Steps .................................................................................... 19  
Best Practices ................................................................................. 21  
  Explaining Access to Users .......................................................... 21  
    Explaining Resource Access ..................................................... 21  
    Explaining Preventive Guardrails ............................................. 22  
  Administrative Tips for Landing Zone Setup ............................... 22  
  Administrative Tips for Landing Zone Maintenance .................... 22  
  Sign in as a Root User ................................................................. 23  
  Recommendations for Setting Up Groups, Roles, and Policies .... 23  
  Guidance for Creating and Modifying AWS Control Tower Resources .................................................. 24  
Managing Resources Outside of AWS Control Tower ..................... 26  
  Referring to Resources Outside of AWS Control Tower .......... 26  
  Changing AWS Control Tower Resource Names ....................... 27  
  Deleting the Core OU ................................................................. 27  
  Changes that Are Updated Automatically .................................. 28  
  Deleting Resources ..................................................................... 28  
Guardrails ....................................................................................... 30  
  Guardrail Behavior and Guidance .............................................. 30  
  Considerations for Guardrails and OUs ...................................... 30  
  Optional Guardrails ................................................................. 31  
  Viewing Guardrail Details .......................................................... 31  
  Enabling Guardrails .................................................................... 32
What Is AWS Control Tower?

AWS Control Tower provides the easiest way to set up and govern a secure, compliant, multi-account AWS environment based on best practices established by working with thousands of enterprises. With AWS Control Tower, end users on your distributed teams can provision new AWS accounts quickly. Meanwhile your central cloud administrators will know that all accounts are aligned with centrally established, company-wide compliance policies.

Features

AWS Control Tower has the following features:

- **Landing zone** – A landing zone is a well-architected, multi-account AWS environment that's based on security and compliance best practices. This is the enterprise-wide container that holds all of your organizational units (OUs), accounts, users, and other resources that you want to be subject to compliance regulation. A landing zone can scale to fit the needs of an enterprise of any size.

- **Guardrails** – A guardrail is a high-level rule that provides ongoing governance for your overall AWS environment. It's expressed in plain language. Two kinds of guardrails exist: preventive and detective. Three categories of guidance apply to the two kinds of guardrails: mandatory, strongly recommended, or elective. For more information about guardrails, see How Guardrails Work (p. 8).

- **Account Factory** – An Account Factory is a configurable account template that helps to standardize the provisioning of new accounts with pre-approved account configurations. AWS Control Tower offers a built-in Account Factory that helps automate the account provisioning workflow in your organization. For more information, see Account Factory (p. 63).

- **Dashboard** – The dashboard offers continuous oversight of your landing zone to your team of central cloud administrators. Use the dashboard to see provisioned accounts across your enterprise, guardrails enabled for policy enforcement, guardrails enabled for continuous detection of policy non-conformance, and noncompliant resources organized by accounts and OUs.

Related Services

AWS Control Tower is built on top of trusted and reliable AWS services including AWS Service Catalog, AWS Single Sign-On, and AWS Organizations. For more information, see Integrated Services (p. 56).

You can incorporate AWS Control Tower with other AWS services into a solution that helps you migrate your existing workloads to AWS. For more information, see How to take advantage of AWS Control Tower and CloudEndure to migrate workloads to AWS.

Pricing

No additional charge exists for using AWS Control Tower. You only pay for the AWS services enabled by AWS Control Tower, and the services you use in your landing zone. For example, you pay for AWS Service Catalog for provisioning accounts with Account Factory, and AWS CloudTrail for events tracked in your landing zone. For information about the pricing and fees associated with AWS Control Tower, see AWS Control Tower pricing.

Are You a First-Time User of AWS Control Tower?

If you’re a first-time user of this service, we recommend that you read the following:
1. If you’re ready to create your first landing zone, see Getting Started with AWS Control Tower (p. 18).
2. For information on drift detection and prevention, see Detecting and Resolving Drift in AWS Control Tower (p. 71).
3. For security details, see Security in AWS Control Tower (p. 77).
4. For information on updating your landing zone and member accounts, see Configuration Update Management in AWS Control Tower (p. 99).

How AWS Control Tower Works

This section describes at a high level how AWS Control Tower works. 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.

Structure of an AWS Control Tower Landing Zone

The structure of a landing zone in AWS Control Tower is as follows:

- **Root** – The parent that contains all other OUs in your landing zone.
- **Core OU** – This OU contains the log archive and audit member accounts. These accounts often are referred to as shared accounts.
- **Custom OU** – The custom OU is created when you launch your landing zone. This and other member OUs contain the member accounts that your users work with to perform their AWS workloads.
- **AWS SSO directory** – This directory houses your AWS SSO users. It defines the scope of permissions for each AWS SSO user.
- **AWS SSO users** – These are the identities that your users can assume to perform their AWS workloads in your landing zone.

What Happens When You Set Up a Landing Zone

When you set up a landing zone, AWS Control Tower performs the following actions in your master account on your behalf:

- Creates three Organizations organizational units (OUs): Root, Core, and Custom.
- Creates two shared accounts: the log archive account and audit account.
- Creates a cloud-native directory in AWS SSO, with preconfigured groups and single sign-on access.
- Applies 20 preventive guardrails to enforce policies.
- Applies six detective guardrails to detect configuration violations.
- Preventive guardrails are not applied to the master account.
- Except for the master account, guardrails are applied to the organization as a whole.

Safely Managing Resources Within Your AWS Control Tower Landing Zone and Accounts

- When you create your landing zone, a number of AWS resources are created. To use AWS Control Tower, you must not modify or delete these AWS Control Tower managed resources outside of the supported methods described in this guide. Deleting or modifying these resources will cause your landing zone to enter an unknown state. For details, see Guidance for Creating and Modifying AWS Control Tower Resources (p. 24)
- When you enable guardrails with strongly recommended guidance, AWS Control Tower creates AWS resources that it manages in your accounts. Do not modify or delete resources created by AWS Control
What Are the Shared Accounts?

In AWS Control Tower, three shared accounts in your landing zone are not provisioned in Account Factory: the master account, the log archive account, and the audit account.

What Is the Master Account?

This is the account that you created specifically for your landing zone. This account is used for billing for everything in your landing zone. It's also used for Account Factory provisioning of accounts, as well as to manage OUs and guardrails.

Note

It is not recommended to run any type of production workloads from an AWS Control Tower master account. Create a separate AWS Control Tower account to run your workloads.

When you set up your landing zone, the following AWS resources are created within your master account.

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Organizations</td>
<td>Accounts</td>
<td>audit</td>
</tr>
<tr>
<td></td>
<td></td>
<td>log archive</td>
</tr>
<tr>
<td>AWS Organizations</td>
<td>OUs</td>
<td>Core</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Custom</td>
</tr>
<tr>
<td>AWS Organizations</td>
<td>Service Control Policies</td>
<td>aws-guardrails-*</td>
</tr>
<tr>
<td>AWS CloudFormation</td>
<td>Stacks</td>
<td>AWSControlTowerBP-BASELINE-CLOUDTRAIL-MASTER</td>
</tr>
<tr>
<td>AWS CloudFormation</td>
<td>StackSets</td>
<td>AWSControlTowerBP-BASELINE-CLOUDTRAIL</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerBP-BASELINE-CLOUDWATCH</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerBP-BASELINE-CONFIG</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerBP-BASELINE-ROLES</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerBP-BASELINE-SERVICE-ROLES</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerBP-BASELINE-SECURITY-ROLES</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerBP-SECURITY-TOPICS</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerGuardrailAWS-GR-AUDIT-BUCKET-PUBLIC-READ-PROHIBITED</td>
</tr>
<tr>
<td>AWS service</td>
<td>Resource type</td>
<td>Resource name</td>
</tr>
<tr>
<td>-----------------------------</td>
<td>---------------</td>
<td>---------------------------------------------------</td>
</tr>
<tr>
<td>AWS Control Tower Guardrail</td>
<td></td>
<td>AWSControlTowerGuardrailAWS-GR-AUDIT-BUCKET-PUBLIC-WRITE-PROHIBITED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerLoggingResources</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerSecurityResources</td>
</tr>
<tr>
<td>AWS Service Catalog</td>
<td>Product</td>
<td>AWS Control Tower Account Factory</td>
</tr>
<tr>
<td>AWS CloudTrail</td>
<td>Trail</td>
<td>aws-controltower-BaselineCloudTrail</td>
</tr>
<tr>
<td>Amazon CloudWatch</td>
<td>CloudWatch Logs</td>
<td>aws-controltower/CloudTrailLogs</td>
</tr>
<tr>
<td>AWS Identity and Access Management</td>
<td>Roles</td>
<td>AWSControlTowerAdmin</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerStackSetRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerCloudTrailRolePolicy</td>
</tr>
<tr>
<td>AWS Identity and Access Management</td>
<td>Policies</td>
<td>AWSControlTowerServiceRolePolicy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerAdminPolicy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerCloudTrailRolePolicy</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerStackSetRolePolicy</td>
</tr>
<tr>
<td>AWS Single Sign-On</td>
<td>Directory groups</td>
<td>AWSAccountFactory</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSAuditAccountAdmins</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerAdmins</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSLogArchiveAdmins</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSLogArchiveViewers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSSecurityAuditors</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSSecurityAuditPowerUsers</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSServiceCatalogAdmins</td>
</tr>
<tr>
<td>AWS Single Sign-On</td>
<td>Permission Sets</td>
<td>AWSAdministratorAccess</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSPowerUserAccess</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSServiceCatalogAdminFullAccess</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSServiceCatalogEndUserAccess</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSReadOnlyAccess</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSOrganizationsFullAccess</td>
</tr>
</tbody>
</table>
What Is the Log Archive Account?

This account works as a repository for logs of API activities and resource configurations from all accounts in the landing zone.

When you set up your landing zone, the following AWS resources are created within your log archive account.

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS CloudFormation</td>
<td>Stacks</td>
<td>StackSet-AWSControlTowerGuardrailAWS-GR-AUDIT-BUCKET-PUBLIC-READ-PROHIBITED-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerGuardrailAWS-GR-AUDIT-BUCKET-PUBLIC-WRITE-PROHIBITED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-CLOUDWATCH-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-CONFIG-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-CLOUDTRAIL-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-SERVICE-ROLES-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-ROLES-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerLoggingResources-</td>
</tr>
<tr>
<td>AWS Config</td>
<td>AWS Config Rules</td>
<td>AWSControlTower_AWS-GR_AUDIT_BUCKET_PUBLIC_READ_PROHIBITED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTower_AWS-GR_AUDIT_BUCKET_PUBLIC_WRITE_PROHIBITED</td>
</tr>
<tr>
<td>AWS CloudTrail</td>
<td>Trails</td>
<td>aws-controltower-BaselineCloudTrail</td>
</tr>
<tr>
<td>Amazon CloudWatch</td>
<td>CloudWatch Event Rules</td>
<td>aws-controltower-ConfigComplianceChangeEventRule</td>
</tr>
<tr>
<td>Amazon CloudWatch</td>
<td>CloudWatch Logs</td>
<td>aws-controltower/CloudTrailLogs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/aws/lambda/aws-controltower-NotificationForwarder</td>
</tr>
<tr>
<td>AWS Identity and Access Management</td>
<td>Roles</td>
<td>aws-controltower-AdministratorExecutionRole</td>
</tr>
</tbody>
</table>
What Are the Shared Accounts?

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource Name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Identity and Access</td>
<td>Policies</td>
<td>AWSControlTowerServiceRolePolicy</td>
</tr>
<tr>
<td>Management</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Amazon Simple Notification</td>
<td>Topics</td>
<td>aws-controltower-SecurityNotifications</td>
</tr>
<tr>
<td>Service</td>
<td></td>
<td></td>
</tr>
<tr>
<td>AWS Lambda</td>
<td>Applications</td>
<td>StackSet-AWSControlTowerBP-BASELINE-CLOUDWATCH-*</td>
</tr>
<tr>
<td>AWS Lambda</td>
<td>Functions</td>
<td>aws-controltower-NotificationForwarder</td>
</tr>
<tr>
<td>Amazon Simple Storage Service</td>
<td>Buckets</td>
<td>aws-controltower-logs-*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-s3-access-logs-*</td>
</tr>
</tbody>
</table>

What Is the Audit Account?

The audit account is a restricted account that’s designed to give your security and compliance teams read and write access to all accounts in your landing zone. From the audit account, you have programmatic access to review accounts, by means of a role that is granted to Lambda functions only. The audit account does not allow you to log in to other accounts manually. For more information about Lambda functions and roles, see Configure a Lambda function to assume a role from another AWS account.

When you set up your landing zone, the following AWS resources are created within your audit account.

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS CloudFormation</td>
<td>Stacks</td>
<td>StackSet-AWSControlTowerGuardrailAWS-GR-AUDIT-BUCKET-PUBLIC-READ-PROHIBITED-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerGuardrailAWS-GR-AUDIT-BUCKET-PUBLIC-WRITE-PROHIBITED-</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-CLOUDWATCH-*</td>
</tr>
</tbody>
</table>
## What Are the Shared Accounts?

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Config</td>
<td>Aggregator</td>
<td>aws-controltower-GuardrailsComplianceAggregator</td>
</tr>
<tr>
<td>AWS Config</td>
<td>AWS Config Rules</td>
<td>AWSControlTower.AWS-GR_AUDIT_BUCKET_PUBLIC_READ_PROHIBITED</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTower.AWS-GR_AUDIT_BUCKET_PUBLIC_WRITE_PROHIBITED</td>
</tr>
<tr>
<td>AWS CloudTrail</td>
<td>Trail</td>
<td>aws-controltower-BaselineCloudTrail</td>
</tr>
<tr>
<td>Amazon CloudWatch</td>
<td>CloudWatch Event Rules</td>
<td>aws-controltower-ConfigComplianceChangeEventRule</td>
</tr>
<tr>
<td>Amazon CloudWatch</td>
<td>CloudWatch Logs</td>
<td>aws-controltower/CloudTrailLogs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/aws/lambda/aws-controltower-NotificationForwarder</td>
</tr>
</tbody>
</table>
### How Guardrails Work

A guardrail is a high-level rule that provides ongoing governance for your overall AWS environment. Each guardrail enforces a single rule, and it's expressed in plain language. Compliance needs evolve, and you can change the elective or strongly recommended guardrails that are in force, at any time, from the AWS Control Tower console. Mandatory guardrails are always applied, and they can't be changed.

Preventive guardrails prevent actions from occurring. For example, the **Disallow policy changes to log archive** guardrail prevents any IAM policy changes within the log archive shared account. Any attempt to perform a prevented action is denied and logged in CloudTrail. The resource is also logged in AWS Config.

Detective guardrails detect specific events when they occur and log the action in CloudTrail. For example, the **Enable encryption for EBS volumes attached to EC2 instances** detects if an unencrypted Amazon EBS volume is attached to an EC2 instance in your landing zone.

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Identity and Access Management</td>
<td>Roles</td>
<td>aws-controltower-AdministratorExecutionRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-CloudWatchLogsRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-ConfigRecorderRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-ForwardSnsNotificationRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-ReadOnlyExecutionRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-SecurityAdministratorRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-SecurityReadOnlyRole</td>
</tr>
<tr>
<td></td>
<td>Policies</td>
<td>AWSControlTowerServiceRolePolicy</td>
</tr>
<tr>
<td>Amazon Simple Notification Service</td>
<td>Topics</td>
<td>aws-controltower-AggregateSecurityNotifications</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-AllConfigNotifications</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-SecurityNotifications</td>
</tr>
<tr>
<td>AWS Lambda</td>
<td>Functions</td>
<td>aws-controltower-NotificationForwarder</td>
</tr>
</tbody>
</table>

---

8
How AWS Regions Work With AWS Control Tower

Currently, AWS Control Tower is supported in the following AWS Regions:

- US East (N. Virginia)
- US East (Ohio)
- US West (Oregon)
- Europe (Ireland)
- Asia Pacific (Sydney)

When you create a landing zone, the region that you're using for access to the AWS Management Console becomes your home AWS Region for AWS Control Tower. During the creation process, some resources are provisioned in the home AWS Region. Other resources, such as OUs and AWS accounts, are global.

Currently, all preventive guardrails work globally. Detective guardrails, however, only work in regions where AWS Control Tower is supported. For more information about the behavior of guardrails when you activate AWS Control Tower in a new region, see Deploying AWS Control Tower to a New AWS Region (p. 101).

How AWS Control Tower Works With Roles to Create and Manage Accounts

AWS Control Tower creates a customer's account by calling the CreateAccount API of AWS Organizations. When AWS Organizations creates this account, it creates a role within that account, which AWS Control Tower names by passing in a parameter to the API. The name of the role is AWSControlTowerExecution.

AWS Control Tower takes over the AWSControlTowerExecution role for all accounts created by Account Factory. Using this role, AWS Control Tower baselines the account and applies mandatory (and any other enabled) guardrails, which results in creation of other roles. These roles in turn are used by other services, such as AWS Config.

**Note**

To baseline an account is to set up its blueprints and guardrails. The baselining process also sets up the centralized logging and security audit roles on the account, as part of deploying the blueprints. AWS Control Tower baselines are contained in the roles that you apply to every managed account.

The AWSControlTowerExecution role, explained

The AWSControlTowerExecution role allows AWS Control Tower to manage your individual accounts and report information about them to your audit and logging accounts.

- AWSControlTowerExecution allows auditing by the AWS Control Tower audit account.
- AWSControlTowerExecution helps you configure your organization's logging, so that all the logs for every account are sent to the logging account.
After you've completed setting up accounts, AWSControlTowerExecution ensures that your selected AWS Control Tower guardrails apply automatically to every individual account in your organization, as well as to every new account you create in AWS Control Tower. Therefore, you can provide compliance and security reports with ease, based on the auditing and logging features embodied by AWS Control Tower guardrails. Your security and compliance teams can verify that all requirements are met, and that no organizational drift has occurred. For more information about drift, see the AWS Control Tower User Guide.

To summarize, the AWSControlTowerExecution role and its associated policy gives you flexible control of security and compliance across your entire organization. Therefore, breaches of security are less likely to occur.

How AWS Control Tower Works With StackSets

AWS Control Tower uses CloudFormation StackSets to set up resources in your accounts. Each stack set has StackInstances that correspond to multiple accounts, and to multiple AWS Regions per account. Control Tower applies updates to certain accounts and AWS Regions selectively, based on certain CloudFormation parameters. When updates are applied to some stack instances, other stack instances may be left in Outdated status. This behavior is expected and normal.

When a stack instance goes into Outdated status, it usually means that the stack corresponding to that stack instance is not aligned with the latest template in the stack set. The stack remains in the older template, so it might not include the latest resources or parameters. The stack is still completely usable.

Here's a quick summary of what behavior to expect, based on AWS CloudFormation parameters that are specified during an update:

If the stack set update includes changes to the template (that is, if the TemplateBody or TemplateURL properties are specified), or if the Parameters property is specified, AWS CloudFormation marks all stack instances with a status of Outdated prior to updating the stack instances in the specified accounts and AWS Regions. If the stack set update does not include changes to the template or parameters, AWS CloudFormation updates the stack instances in the specified accounts and Regions, while leaving all other stack instances with their existing stack instance status. To update all of the stack instances associated with a stack set, do not specify the Accounts or Regions properties.

For more information, see Update Your Stack Set in the AWS CloudFormation User Guide.

AWS Control Tower and VPCs

This section is intended primarily for network administrators. Your company's network administrator usually is the person who selects the overall CIDR range for your AWS Control Tower organization. The network administrator then allocates subnets from within that range for specific purposes.

Here are some essential facts about AWS Control Tower VPCs:

- The VPC created by AWS Control Tower when you provision an account in Account Factory is not the same as the AWS default VPC.
- When AWS Control Tower sets up a new account in a supported AWS Region, AWS Control Tower automatically deletes the default AWS VPC, and it sets up a new VPC configured by AWS Control Tower.
- Each AWS Control Tower account is allowed one VPC that's created by AWS Control Tower. An account can have additional AWS VPCs within the account limit.
- Every AWS Control Tower VPC has three Availability Zones. By default, each Availability Zone is assigned one public subnet and two private subnets. Therefore, each AWS Control Tower VPC contains nine subnets by default, divided into three Availability Zones.
• Each of the nine subnets in your AWS Control Tower VPC is assigned a unique range, of equal size.
• The number of subnets in a VPC is configurable. For more information about how to change your VPC subnet configuration, see the Account Factory topic.
• Because the IP addresses do not overlap, the nine subnets within your AWS Control Tower VPC can communicate with each other in an unrestricted manner.

If the default configuration or capabilities of the AWS Control Tower VPC do not meet your needs, you can use other AWS services to configure your VPC. For more information about how to work with VPCs and AWS Control Tower see Building a Scalable and Secure Multi-VPC AWS Network Infrastructure.

Note
If you set the Account Factory VPC configuration so that public subnets are enabled when provisioning a new account, Account Factory configures VPC to create a NAT Gateway. You will be billed for your usage by Amazon VPC.

CIDR and Peering for VPC and AWS Control Tower

When you choose a CIDR range for your VPC, AWS Control Tower validates the IP address ranges according to the RFC 1918 specification. Account Factory allows a CIDR block of up to /16 in the ranges of:

• 10.0.0.0/8
• 172.16.0.0/12
• 192.168.0.0/16
• 100.64.0.0/10 (only if your internet provider allows usage of this range)

The /16 delimiter allows up to 65,536 distinct IP addresses.

You can assign any valid IP addresses from the following ranges:

• 10.0.x.x to 10.255.x.x
• 172.16.x.x – 172.31.x.x
• 192.168.0.0 – 192.168.255.255 (no IPs outside of 192.168 range)

If the range you specify is outside of these, AWS Control Tower provides an error message.

The default CIDR range is 172.31.0.0/16.

When AWS Control Tower creates a VPC using the CIDR range you select, it assigns the identical CIDR range to every VPC for every account you create within the organizational unit (OU). Due to the default overlap of IP addresses, this implementation does not initially permit peering among any of your AWS Control Tower VPCs in the OU.

Within each VPC, AWS Control Tower divides your specified CIDR range evenly into nine subnets. None of the subnets within a VPC overlap. Therefore, they all can communicate with each other, within the VPC.

The best practice for controlling communication among your VPC subnets, if needed, is to set up access control lists with rules that define the permitted traffic flow. Use security groups for control of traffic among specific instances.

In summary, by default, subnet communication within the VPC is unrestricted. By default, all AWS Control Tower VPCs have the same default CIDR range. For communication across multiple VPCs, AWS Control Tower does not restrict VPC-to-VPC peering, but it does not provide an interface designed for setting it up from the AWS Control Tower console. The next section describes some options for VPC peering.
Options for VPC Peering in AWS Control Tower

AWS Control Tower offers VPC endpoint services through AWS PrivateLink as a possible solution for VPC peering among AWS Control Tower VPCs. With AWS PrivateLink, packets can be sent directly from a specific IP address in one VPC to another specific IP address within another VPC.

However, another peering option is available within AWS Control Tower. If you change the CIDR range in the settings of Account Factory, all new accounts that are subsequently created by AWS Control Tower (using Account Factory) are assigned the new CIDR range. The old accounts are not updated. For example, you can create an account, then change the CIDR range and create a new account, and the VPCs allocated to those two accounts can be peered. Peering is possible because their IP address ranges are not identical. For information about how to change account settings, see the Account Factory documentation on updating an account.

Notes on VPC and CIDR

Some network administrators may realize that it is possible to peer two subnets in two different VPCs (that is, in two different accounts) without changing the CIDR settings for accounts. Because the nine subnets in a VPC do not overlap, peering technically is possible from \([\text{VPC1, subnet 1}]\) to \([\text{VPC2, subnet 2}]\), for example. However, this approach depends on an implementation detail of how subnet ranges are allocated within a VPC. We don’t recommend this method of peering, because it could fail at any time.

When working with VPCs, AWS Control Tower makes no distinction at the Region level. Every subnet is allocated from the exact CIDR range that you specify. The VPC subnets can exist in any Region.
Plan Your AWS Control Tower Landing Zone

When you go through the setup process, AWS Control Tower launches a resource associated with your account, called a landing zone, which serves as a home for your organizations and their accounts.

**Note**
You can have one landing zone per organization.

**Ways to Set Up AWS Control Tower**

- Launching AWS Control Tower in an Existing Organization (p. 13): This section is for customers who have existing AWS Organizations ready to bring into governance by AWS Control Tower.
- Launching AWS Control Tower in a New Organization (p. 14): This section is for customers without existing AWS Organizations, OUs, and accounts.

**Launching AWS Control Tower in an Existing Organization**

By setting up an AWS Control Tower landing zone in an existing organization, you can start working immediately, in parallel with your existing AWS Organizations environment. Your other OUs created within AWS Organizations are unchanged, because they are not managed by AWS Control Tower. You can continue to use those OUs and accounts exactly as they are.

AWS Control Tower consolidates by using the master account from your existing organization as its master account. No new master account is needed.

**Note**
To set up AWS Control Tower on an existing organization, your service limits must allow for the creation of at least two additional accounts.

**Effects of AWS Control Tower on your existing organization**

AWS Control Tower adds two accounts to your organization: an audit account and a logging account. These accounts keep a record of actions taken by your team, in their individual user accounts. The audit and logging accounts appear in the Core OU within your AWS Control Tower landing zone.

When you set up your landing zone, the accounts added by AWS Control Tower become part of your existing AWS Organizations, and as such they become part of the billing for your existing organization.

**Summary of Capabilities**

Enabling AWS Control Tower on an existing AWS Organizations organization provides several major enhancements to the organization.

- It allows for unified billing across your organization's groups, because accounts added by AWS Control Tower will become part of your existing organization.
- It gives you the ability to administer all accounts from one master account in your OU.
• It simplifies how you apply and enforce guardrails that cover security and compliance for existing and new accounts.

**Important**

Launching your AWS Control Tower landing zone in an existing AWS Organizations organization does not enable you to extend AWS Control Tower governance from that organization to other OUs or accounts that are not registered with AWS Control Tower.

To launch AWS Control Tower in your existing organization, follow the process outlined in Getting Started with AWS Control Tower (p. 18).

For more information about how AWS Control Tower interacts with existing AWS Organizations, see Enable AWS Control Tower on an Existing Organization (p. 113).

---

### Launching AWS Control Tower in a New Organization

If you're new to AWS Control Tower and you haven't worked with AWS Organizations, the best place to begin is with our Setting Up (p. 15) document.

AWS Control Tower sets up an organization for you automatically when you don't have one set up.
Setting Up

Before you use AWS Control Tower for the first time, complete the following tasks:

1. Sign up for AWS (p. 15)
2. Create an IAM User (p. 15)

These tasks create an AWS account and an IAM user with administrator privileges for the account. For information on additional setup tasks specifically for AWS Control Tower, see Getting Started with AWS Control Tower (p. 18).

Sign up for AWS

When you sign up for Amazon Web Services (AWS), your AWS account is automatically signed up for all services in AWS, including AWS Control Tower. If you have an AWS account already, skip to the next task. If you don’t have an AWS account, use the following procedure to create one.

To create an AWS account

2. Follow the online instructions.

Part of the sign-up procedure involves receiving a phone call and entering a verification code on the phone keypad.

Note your AWS account number, because you need it for the next task.

Create an IAM User

Services in AWS, such as AWS Control Tower, require that your user account must provide credentials, so that the service can determine whether you have permission to utilize its resources. AWS recommends that you don't make requests to other services from the root user credentials of your AWS account. Instead, create an AWS Identity and Access Management (IAM) user and grant that user full access. We call these full-access users administrators.

You can use the administrator credentials, instead of AWS account root user credentials of your account, to interact with AWS and perform tasks, such as create users and grant them the appropriate permissions. For more information, see Root Account Credentials vs. IAM User Credentials in the AWS General Reference and IAM Best Practices in the IAM User Guide.

If you signed up for AWS but have not created an IAM user for yourself, you can create one using the IAM Management Console.

To create an administrator user for yourself and add the user to an administrators group (console)

1. Sign in to the IAM console as the account owner by choosing Root user and entering your AWS account email address. On the next page, enter your password.
Create an IAM User

2. In the navigation pane, choose Users and then choose Add user.
3. For User name, enter Administrator.
4. Select the check box next to AWS Management Console access. Then select Custom password, and then enter your new password in the text box.
5. (Optional) By default, AWS requires the new user to create a new password when first signing in. You can clear the check box next to User must create a new password at next sign-in to allow the new user to reset their password after they sign in.
6. Choose Next: Permissions.
7. Under Set permissions, choose Add user to group.
8. Choose Create group.
9. In the Create group dialog box, for Group name enter Administrators.
10. Choose Filter policies, and then select AWS managed -job function to filter the table contents.
11. In the policy list, select the check box for AdministratorAccess. Then choose Create group.

You must activate IAM user and role access to Billing before you can use the AdministratorAccess permissions to access the AWS Billing and Cost Management console. To do this, follow the instructions in step 1 of the tutorial about delegating access to the billing console.

12. Back in the list of groups, select the check box for your new group. Choose Refresh if necessary to see the group in the list.
13. Choose Next: Tags.
14. (Optional) Add metadata to the user by attaching tags as key-value pairs. For more information about using tags in IAM, see Tagging IAM Entities in the IAM User Guide.
15. Choose Next: Review to see the list of group memberships to be added to the new user. When you are ready to proceed, choose Create user.

You can use this same process to create more groups and users and to give your users access to your AWS account resources. To learn about using policies that restrict user permissions to specific AWS resources, see Access Management and Example Policies.

To sign in as this new IAM user, first sign out of the AWS Management Console. Then use the following URL, where your_aws_account_id is your AWS account number without the hyphens (for example, if your AWS account number is 1234-5678-9012, your AWS account ID is 123456789012).

```
https://your_aws_account_id.signin.aws.amazon.com/console/
```

Enter the IAM user name and password that you just created. When you're signed in, the navigation bar displays your_user_name@your_aws_account_id.

If you don't want the URL for your sign-in page to contain your AWS account ID, you can create an account alias. To do so, from the IAM dashboard, choose Create Account Alias and enter an alias, such as your company name. To sign in after you create an account alias, use the following URL.

```
https://your_account_alias.signin.aws.amazon.com/console/
```

To verify the sign-in link for IAM users for your account, open the IAM console and check under AWS Account Alias on the dashboard.

Note
We strongly recommend that you adhere to the best practice of using the Administrator IAM user below and securely lock away the root user credentials. Sign in as the root user only to perform a few account and service management tasks.

Note
You must activate IAM user and role access to Billing before you can use the AdministratorAccess permissions to access the AWS Billing and Cost Management console. To do this, follow the instructions in step 1 of the tutorial about delegating access to the billing console.
Set up MFA

Because of the nature of AWS Control Tower, we strongly recommend that you enable multi-factor authentication (MFA) for your master account. For more information, see Enable MFA on the AWS Account Root User in the IAM User Guide.

Next Step

Getting Started with AWS Control Tower (p. 18)
Getting Started with AWS Control Tower

This is the AWS Control Tower getting started procedure for central cloud administrators. Use this procedure when you’re ready to set up your landing zone. From start to finish, it should take about an hour. This procedure has a prerequisite and two steps.

Prerequisite: Automated Pre-Launch Checks for Your Master Account

Before AWS Control Tower sets up the landing zone, it automatically runs a series of pre-launch checks in your account. There's no action required on your part for these checks, which ensure that your master account is ready for the changes that establish your landing zone. Here are the checks that AWS Control Tower runs before setting up a landing zone:

- The existing service limits for the AWS account must be sufficient for AWS Control Tower to launch. For more information, see Limitations and Quotas in AWS Control Tower (p. 103).
- The AWS account must be subscribed to the following AWS services:
  - Amazon Simple Storage Service (Amazon S3)
  - Amazon Elastic Compute Cloud (Amazon EC2)
  - Amazon SNS
  - Amazon Virtual Private Cloud (Amazon VPC)
  - AWS CloudFormation
  - AWS CloudTrail
  - Amazon CloudWatch
  - AWS Config
  - AWS Identity and Access Management (IAM)
  - AWS Lambda

  **Note**
  By default, all accounts are subscribed to these services.

- If AWS Single Sign-On (AWS SSO) is already set up, the AWS Control Tower home region must be the same as the AWS SSO region.
- The AWS account cannot have trusted access enabled in the organization master account for either AWS Config or AWS CloudTrail. We recommend that you do not turn AWS Config off to set up AWS Control Tower and then turn it back on. If you do so, you'll incur additional charges.

Step One: Create Your Shared Account Email Addresses

If you’re setting up your landing zone in a new AWS account, for information on creating your account and your IAM administrator, see Setting Up (p. 15).
To set up your landing zone, AWS Control Tower requires two unique email addresses that aren't already associated with an AWS account. These email addresses should each be a collaborative inbox, a shared email account for the different users in your enterprise that will do specific work related to AWS Control Tower. The email addresses are:

- **Audit account** – This account is for your team of users that need access to the audit information made available by AWS Control Tower. You can also use this account as the access point for third-party tools that will perform programmatic auditing of your environment to help you audit for compliance purposes.
- **Log archive account** – This account is for your team of users that need access to all the logging information for all of your managed accounts within managed OUs in your landing zone.

### Step Two: Set Up Your Landing Zone

Before you set up your AWS Control Tower landing zone, determine the most appropriate home region. For more information, see [Administrative Tips for Landing Zone Setup](p. 22).

AWS Control Tower has no APIs or programmatic access. To set up your landing zone, perform the following procedure:

**To set up your landing zone**

2. In the console, verify that you are working in your desired home region for AWS Control Tower. Then choose **Set up your landing zone**.
3. Provide the email addresses for your log archive and audit accounts. Note that the email addresses must not already have associated AWS accounts.
4. Review the **Service permissions**, and when you're ready, choose **I understand the permissions AWS Control Tower will use to administer AWS resources and enforce rules on my behalf**.
5. Choose **Launch your AWS Control Tower**.

This starts the process of setting up your landing zone, which can take about an hour to complete. During setup, your core accounts are created, your root and Core OUs are created, and AWS resources are created, modified, or deleted.

**Important**

The email address you provided for the audit account will receive **AWS Notification - Subscription Confirmation** emails from every AWS Region supported by AWS Control Tower. To receive compliance emails in your audit account, you must choose the **Confirm subscription** link within each email from each AWS Region supported by AWS Control Tower.

### Next Steps

Now that your landing zone is set up, it's ready for use.

To learn more about how you can use AWS Control Tower, see the following topics:

- For recommended administrative practices, see [Best Practices](p. 23).
- You can set up AWS SSO users and groups with specific roles and permissions. For recommendations, see [Recommendations for Setting Up Groups, Roles, and Policies](p. 23).
- Your end users can provision their own AWS accounts in your landing zone using Account Factory. For more information, see [Permissions for Configuring and Provisioning Accounts](p. 63).
• To assure Compliance Validation for AWS Control Tower (p. 98), your central cloud administrators can review log archives in the log archive account, and designated third-party auditors can review audit information in the audit shared account.

• From time to time, you may need to update your landing zone to get the latest backend updates, the latest guardrails, and to keep your landing zone up-to-date. For more information, see Configuration Update Management in AWS Control Tower (p. 99).

• If you encounter issues while using AWS Control Tower, see Troubleshooting (p. 122).
Best Practices for Account Administrators

This topic is intended primarily for master account administrators.

Master account administrators are responsible for explaining some tasks that AWS Control Tower guardrails prevent their member account administrators from doing. This topic describes some best practices and procedures for transferring this knowledge, and it gives other tips for setting up and maintaining your AWS Control Tower environment efficiently.

Explaining Access to Users

The AWS Control Tower console is available only to users with the master account administrator permissions. Only these users can perform administrative work within your landing zone. In accordance with best practices, this means that the majority of your users and member account administrators will never see the AWS Control Tower console. As a member of the master account administrator group, it's your responsibility to explain the following information to the users and administrators of your member accounts, as appropriate.

- Explain which AWS resources that users and administrators have access to within the landing zone.
- List the preventive guardrails that apply to each Organizational Unit (OU) so that the other administrators can plan and execute their AWS workloads accordingly.

Explaining Resource Access

Some administrators and other users may need an explanation of the AWS resources to which they have access to within your landing zone. This access can include programmatic access and console-based access. Generally speaking, read access and write access for AWS resources is allowed. To perform work within AWS, your users require some level of access to the specific services they need to do their jobs.

Some users, such as your AWS developers, may need to know about the resources to which they have access, so they can create engineering solutions. Other users, such as the end users of the applications that run on AWS services, do not need to know about AWS resources within your landing zone.

AWS offers tools to identify the scope of a user’s AWS resource access. After you identify the scope of a user’s access, you can share that information with the user, in accordance with your organization’s information management policies. For more information about these tools, see the links that follow.

- **AWS access advisor** – The AWS Identity and Access Management (IAM) access advisor tool lets you determine the permissions that your developers have by analyzing the last timestamp when an IAM entity, such as a user, role, or group, called an AWS service. You can audit service access and remove unnecessary permissions, and you can automate the process if needed. For more information, see our [AWS Security blog post](https://aws.amazon.com/blogs/security/).  

- **IAM policy simulator** – With the IAM policy simulator, you can test and troubleshoot IAM-based and resource-based policies. For more information, see [Testing IAM Policies with the IAM Policy Simulator](https://docs.aws.amazon.com/IAM/latest/UserGuide/id_sims.html).

- **AWS CloudTrail logs** – You can review AWS CloudTrail logs to see actions taken by a user, role, or AWS service. For more information about CloudTrail, see the [AWS CloudTrail User Guide](https://docs.aws.amazon.com/AmazonCloudTrail/latest/UserGuide/).

Actions taken by CloudTrail landing zone administrators are logged in the landing zone master account. Actions taken by member account administrators and users are logged in the shared log archive account.
You can view a summary table of AWS Control Tower events in the Activities page.

Explaining Preventive Guardrails

A preventive guardrail ensures that your organization's accounts maintain compliance with your corporate policies. The status of a preventive guardrail is either enforced or not-enabled. A preventive guardrail prevents policy violations by using service control policies and AWS Lambda functions. In comparison, a detective guardrail only informs you of various events or states that exist.

Some of your users, such as AWS developers, may need to know about the preventive guardrails that apply to any accounts and OUs they use, so they can create engineering solutions. The following procedure offers some guidance on how to provide this information for the right users, according to your organization's information management policies.

Note
This procedure assumes you've already created at least one child OU within your landing zone, as well as at least one AWS Single Sign-On user.

To show preventive guardrails for users with a need to know

2. From the left navigation, choose Organizational units.
3. From the table, choose the name of one of the OUs for which your user needs information about the applicable guardrails.
4. Note the name of the OU and the guardrails that apply to this OU.
5. Repeat the previous two steps for each OU about which your user needs information.

For detailed information about the guardrails and their functions, see Guardrails in AWS Control Tower (p. 30).

Administrative Tips for Landing Zone Setup

• The AWS Region where you do the most work should be your home region.
• Set up your landing zone and deploy your Account Factory accounts from within your home region.
• If you’re investing in several AWS Regions, be sure that your cloud resources are in the region where you’ll do most of your cloud administrative work and run your workloads.
• The audit and other buckets are created in the same AWS Region from which you launch AWS Control Tower. We recommend that you do not move these buckets.
• When launching, AWS STS endpoints must be activated in the master account, for all regions supported by AWS Control Tower. Otherwise, the launch may fail midway through the configuration process.

Administrative Tips for Landing Zone Maintenance

• You can make your own log buckets in the log archive account. Just be sure to leave the buckets created by AWS Control Tower. Note that your Amazon S3 access logs must be in the same AWS Region as the source buckets.
• By keeping your workloads and logs in the same AWS Region, you reduce the cost that would be associated with moving and retrieving log information across regions.
The VPC created by AWS Control Tower is limited to the AWS Regions in which AWS Control Tower is available. Some customers whose workloads run in non-supported regions may want to disable the VPC that is created with your Account Factory account. They may prefer to create a new VPC using the AWS Service Catalog portfolio, or to create a custom VPC that runs in only the required regions.

The VPC created by AWS Control Tower is not the same as the default VPC that is created for all AWS accounts. In regions where AWS Control Tower is supported, AWS Control Tower deletes the default AWS VPC when it creates the AWS Control Tower VPC.

If you delete your default VPC in your home AWS Region, it's best to delete it in all other AWS Regions.

Sign in as a Root User

Certain administrative tasks require that you must sign in as a root user. You can sign in as a root user to an AWS account that was created by account factory in AWS Control Tower.

You must sign in as a root user to perform the following actions:

- Change certain account settings, including the account name, root user password, or email address. For more information, see Updating and Moving Account Factory Accounts (p. 66).
- To change or enable your AWS Support plan.
- To close an AWS Account.
- For more information about actions that require root login credentials, please see AWS Tasks that Require AWS Root Login Credentials.

To sign in as root user

1. Open the AWS sign-in page.

   If you don't have the email address of the AWS account to which you require access, you can get it from AWS Control Tower. Open the console for the master account, choose Accounts, and look for the email address.

2. Enter the email address of the AWS account to which you require access, and then choose Next.

3. Choose Forgot password? to have password reset instructions sent to the root user email address.

4. Open the password reset email message from the root user mailbox, then follow the instructions to reset your password.

5. Open the AWS sign-in page, then sign in with your reset password.

Recommendations for Setting Up Groups, Roles, and Policies

As you set up your landing zone, it's a good idea to decide ahead of time which users will require access to certain accounts and why. For example, a security account should be accessible only to the security team, the master account should be accessible only to the cloud administrators' team, and so forth.

Recommended Restrictions

You can restrict the scope of administrative access to your organizations by setting up an IAM role or policy that allows administrators to manage AWS Control Tower actions only. The recommended approach is to use the IAM Policy `arn:aws:iam::aws:policy/service-role/AWSControlTowerServiceRolePolicy`. With the `AWSControlTowerServiceRolePolicy` role
enabled, an administrator can manage AWS Control Tower only. Be sure to include appropriate access to AWS Organizations for managing your preventive guardrails, and SCPs, and access to AWS Config, for managing detective guardrails, in each account.

When you're setting up the shared audit account in your landing zone, we recommend that you assign the AWSSecurityAuditors group to any third-party auditors of your accounts. This group gives its members read-only permission. An account must not have write permissions on the environment that it is auditing, because it can violate compliance with Separation of Duty requirements for auditors.

Guidance for Creating and Modifying AWS Control Tower Resources

We recommend the following practices as you create and modify resources in AWS Control Tower. This guidance might change as the service is updated.

General Guidance

- Do not modify or delete resources created by AWS Control Tower in the master account or in the shared accounts. Modification of these resources can require an update to your landing zone.
- Do not modify or delete the AWS Identity and Access Management (IAM) roles created within the shared accounts in the core organizational unit (OU). Modification of these roles can require an update to your landing zone.
- For more information about the resources created by AWS Control Tower, see What Are the Shared Accounts? (p. 3)
- Do not disallow usage of any AWS Regions through either SCPs or AWS STS. Doing so will break AWS Control Tower.
- In general, AWS Control Tower performs a single action at a time, which must be completed before another action can begin. For example, if you attempt to provision an account while the process of enabling a guardrail is already in operation, account provisioning will fail.

AWS Organizations Guidance

- Do not use AWS Organizations to update service control policies (SCPs) attached to an OU that is registered with AWS Control Tower. Doing so could result in the guardrails entering an unknown state, which will require you to re-enable affected guardrails in AWS Control Tower. Instead, you can create new SCPs and attach those to the OUs rather than editing the SCPs that AWS Control Tower has created.
- Moving individual accounts into AWS Control Tower, from outside of a registered OU, causes drift that must be repaired. See Types of Governance Drift (p. 72).
- If you use AWS Organizations to create, invite, or move accounts within an organization registered with AWS Control Tower, those accounts are not enrolled by AWS Control Tower and those changes are not recorded. If you need access to these accounts through SSO, see Member Account Access.
- If you use AWS Organizations to move an OU into an organization created by AWS Control Tower, the external OU is not registered by AWS Control Tower.
- Nested OUs are not accessible in AWS Control Tower, because AWS Control Tower displays only the top-level OUs.
- If you use AWS Organizations to rename an account or OU that was created by AWS Control Tower, you must repair your landing zone so that the new name is displayed by AWS Control Tower.
- If you use AWS Organizations to delete an OU that was created by AWS Control Tower, you also must delete the OU in AWS Control Tower. It cannot be used to contain accounts. You will not be able to provision a new account to this OU using Account Factory.
AWS Single Sign-On Guidance

- For more information about how the behavior of AWS Control Tower interacts with AWS SSO and different identity sources, refer to Considerations for Changing Your Identity Source in the AWS SSO documentation.
- See Managing Users and Access Through AWS Single Sign-On (p. 58) for more information about working with AWS Control Tower and AWS SSO.

Account Factory Guidance

- When you use Account Factory to provision new accounts in AWS Service Catalog, do not define TagOptions, enable notifications, or create a provisioned product plan. Doing so can result in a failure to provision a new account.
- If you are authenticated as an IAM user when you provision accounts in Account Factory or when you use the Enroll account feature, be sure the IAM user is added to the AWS Service Catalog portfolio so that it has the correct permissions. Otherwise, you may receive an error message from AWS Service Catalog that is difficult to understand. Common causes for this type of error are given in the Troubleshooting guide. In particular, refer to the section entitled No Launch Paths Found Error (p. 125).
- Remember that only one account can be provisioned at a time.
Managing Resources Outside of AWS Control Tower

AWS Control Tower sets up accounts, organizational units, and other resources on your behalf, but you are the owner of these resources. You can change these resources within AWS Control Tower or outside it. The most common place to change resources outside of AWS Control Tower is the AWS Organizations console. This topic describes how to reconcile changes to AWS Control Tower resources when you make the changes outside of AWS Control Tower.

Renaming, deleting, and moving resources outside of the AWS Control Tower console can cause the console to become out of sync and display outdated information. Some changes can be reconciled automatically. Other changes require a repair to your landing zone to update the information that's displayed in the AWS Control Tower console.

Tasks that require landing zone repair to reconcile your changes

- Renaming a registered organizational unit (OU)
- Renaming an enrolled account
- Deleting the Core OU (A special case, not to be done lightly.)

Changes that are updated automatically by AWS Control Tower

- Changing the email address of an enrolled account
- Deleting a registered OU (Except the Core OU, which requires an update.)
- Deleting an enrolled account

Note

AWS Service Catalog handles changes differently than AWS Control Tower. AWS Service Catalog may create a change in governance posture when it reconciles your changes. For more information about updating a provisioned product, see Updating Provisioned Products in the AWS Service Catalog documentation.

In general, changes that you make outside the AWS Control Tower console to AWS Control Tower resources create a state of repairable drift in your landing zone. For more information about these changes, see Repairable Changes to Resources (p. 71).

Referring to Resources Outside of AWS Control Tower

When you create new OUs and accounts outside of AWS Control Tower, they are not governed by AWS Control Tower, even though they may be displayed.

Creating an OU

Organizational Units (OUs) created outside of AWS Control Tower are referred to as Unregistered. They are displayed in the OU list page, but they are not governed by AWS Control Tower guardrails.
Creating an account

Accounts created outside of AWS Control Tower are referred to as *Unenrolled*. Accounts that belong to an organization that’s registered with AWS Control Tower are displayed in the **Accounts list** page. Accounts that do not belong to a registered organization can be invited by using the AWS Organizations console. This invitation to join does not enroll the account in AWS Control Tower or extend AWS Control Tower governance to the account. To extend governance by enrolling the account, go to the Account Factory page in AWS Control Tower and choose **Enroll account**.

Changing AWS Control Tower Resource Names

You can change the names of your organizational units (OUs) and accounts outside of the AWS Control Tower console, but you must also repair your landing zone so that you can see the updates that you've made.

Renaming an OU

In AWS Organizations, you can change the name of an OU by using either the API or the console. When you change an OU name outside of AWS Control Tower, you also must repair your landing zone to ensure that AWS Control Tower stays consistent with AWS Organizations. The **Repair** workflow ensures consistency across services for the Core and non-Core (workload) OU names. You can repair this type of drift from the **Settings** page. See “Resolving Drift” in Detecting and Resolving Drift in AWS Control Tower (p. 71).

AWS Control Tower displays the names of OUs in the console and in Account Factory, and you can see when your landing zone repair has succeeded.

Renaming an enrolled account

Each AWS account has a display name that can be changed in the AWS Billing and Cost Management console. When you change an account name, you also must repair your AWS Control Tower landing zone to ensure data consistency between services for Core and non-Core (workload) account names.

**Note**
The provisioned product name in AWS Control Tower is set to the AWS account name when the account is provisioned. If you change this value, you also must update the AWS Service Catalog provisioned product. This change in AWS Service Catalog may cause a change in governance posture for the account.

Deleting the Core OU

Before you can delete the **Core** OU, you must make sure it contains no accounts. Specifically, you must remove the log archive and audit accounts from the OU. We recommend that you move these accounts to another OU. When you run the **Repair** function, AWS Control Tower creates a new Core OU, and it moves these accounts back into the new Core OU. AWS Control Tower marks these accounts as drifted.

**Note**
The action of deleting your Core OU is not to be performed without due consideration. The action could create compliance concerns if logging is suspended temporarily, and because some guardrails might not be enforced.

If you delete the **Core** OU, you will see an error message prompting you to repair your landing zone. You must repair your landing zone before you can take any other actions in AWS Control Tower. You will not be able to perform any actions in the AWS Control Tower console and you will not be able to create any new accounts in AWS Service Catalog until the repair is done.
This type of drift is a special case. You won’t be able to view the Settings page to see the Repair button there. In this situation, the landing zone repair process creates a new Core OU and moves the two shared accounts into the new Core OU. The same process repairs the drift in the log archive and audit accounts.

For general information about drift, see "Resolving Drift" in Detecting and Resolving Drift in AWS Control Tower (p. 71).

Changes that Are Updated Automatically

Changes that you make to your account email addresses are updated by AWS Control Tower automatically, but Account Factory does not update them automatically.

Changing the email address of a governed account

AWS Control Tower retrieves and displays email addresses as required by the console experience. Therefore, core and non-core account email addresses are updated and shown consistently in AWS Control Tower after you change them.

Note
In AWS Service Catalog, the Account Factory displays the parameters that were specified in the console when you created a provisioned product. However, the original account email address is not updated automatically when the account email address changes. That’s because the account is conceptually contained within the provisioned product; it is not the same as the provisioned product. To update this value, you must update the provisioned product, which may cause a change in governance posture.

Deleting Resources

You can delete OUs and accounts in AWS Control Tower and you don't need to take any further action to see the updates. Account Factory is updated automatically when you delete an OU, but not when you delete an account.

Deleting a registered OU (except the Core OU)

Within AWS Organizations, you can remove empty organizational units (OUs) by using the API or the console. From the AWS Control Tower console, you can choose Deregister OU to remove non-core, governed OUs that are empty. OUs that contain accounts cannot be deleted.

AWS Control Tower receives a notification from AWS Organizations when an OU is deleted. It updates the OU list in the Account Factory, so that the list of registered OUs remains consistent.

If you see a deleted OU displayed in the AWS Control Tower console, repair your landing zone to remove outdated entries. You also can choose Deregister OU in the AWS Control Tower console to clean up stale entries.

Note
In AWS Service Catalog, the Account Factory is updated to remove the deleted OU from the list of available OUs into which you can provision an account.

Deleting an enrolled account from an OU

When you remove an enrolled, non-core account, AWS Control Tower receives a notification and makes updates, so that the information remains consistent.

If you see a deleted account displayed in the AWS Control Tower console, repair your landing zone to remove the outdated entry.
Note
In AWS Service Catalog, the Account Factory provisioned product that represents the governed account is not updated to remove the account. Instead, the provisioned product is displayed as Tainted and in an error state. To clean up, go to AWS Service Catalog, choose the provisioned product, and then choose Terminate.
Guardrails in AWS Control Tower

A guardrail is a high-level rule that provides ongoing governance for your overall AWS environment. It's expressed in plain language. Through guardrails, AWS Control Tower implements preventive or detective controls that help you govern your resources and monitor compliance across groups of AWS accounts.

A guardrail applies to an entire organizational unit (OU), and every AWS account within the OU is affected by the guardrail. Therefore, when users perform work in any AWS account in your landing zone, they're always subject to the guardrails that are governing their account's OU.

The purpose of guardrails

Guardrails enable you to express your policy intentions. For example, if you enable the detective Disallow public read access to S3 buckets guardrail on an OU, you can determine whether a user has attempted public read access to any S3 buckets for any accounts under that OU.

Guardrail Behavior and Guidance

Guardrails are categorized according to their behavior and their guidance.

The behavior of each guardrail is either preventive or detective. Guardrail guidance refers to the recommended practice for how to apply each guardrail to your OUs. The guidance of a guardrail is independent of whether its behavior is preventive or detective.

Guardrail behavior

- **Prevention** – A preventive guardrail ensures that your accounts maintain compliance, because it disallows actions that lead to policy violations. The status of a preventive guardrail is either enforced or not enabled. Preventive guardrails are supported in all AWS Regions.
- **Detection** – A detective guardrail detects noncompliance of resources within your accounts, such as policy violations, and provides alerts through the dashboard. The status of a detective guardrail is either clear, in violation, or not enabled. Detective guardrails apply only in those AWS Regions supported by AWS Control Tower.

Implementation of guardrail behavior

- The preventive guardrails are implemented using Service Control Policies (SCPs), which are part of AWS Organizations.
- The detective guardrails are implemented using AWS Config rules and AWS Lambda functions.
- Certain mandatory guardrails are implemented by means of a single SCP that performs multiple actions, rather than as unique SCPs. Therefore, the same SCP is shown in the guardrail reference, under each mandatory guardrail to which that SCP applies.

Guardrail guidance

AWS Control Tower provides three categories of guidance: mandatory, strongly recommended, and elective guardrails.

- Mandatory guardrails are always enforced.
- Strongly recommended guardrails are designed to enforce some common best practices for well-architected, multi-account environments.
• Elective guardrails enable you to track or lock down actions that are commonly restricted in an AWS enterprise environment.

**Defaults:** When you create a new landing zone, all mandatory guardrails are enabled by default. Strongly recommended and elective guardrails are not enabled by default.

### Considerations for Guardrails and OUs

When working with guardrails and OUs, consider the following properties:

• When you enable guardrails on an organizational unit, those guardrails apply to all child accounts under the OU.
• After you create your landing zone, all resources in your landing zone, for example, S3 buckets, are subject to guardrails.
• OUs created through AWS Control Tower have mandatory guardrails applied to them automatically, and other guardrails applied at the discretion of administrators. In contrast, OUs created outside of an AWS Control Tower landing zone can't have guardrails applied to them. These *ungoverned* OUs are not displayed in the AWS Control Tower console.
• Accounts created through Account Factory inherit their parent OU's guardrails. Accounts created outside of a landing zone do not inherit guardrails, and these *ungoverned* accounts are not displayed in the AWS Control Tower console.

**Exceptions to guardrails**

• The root user and any IAM administrators in the master account can perform work that guardrails would otherwise deny. This exception is intentional. It prevents the master account from entering into an unusable state. All actions taken within the master account continue to be tracked in the logs contained within the log archive account, for purposes of accountability and auditing.

### Optional Guardrails

Strongly recommended and elective guardrails are optional, which means that you can customize the level of enforcement for your landing zone by choosing which ones to enable. Optional guardrails are not enabled by default. For more information about optional guardrails, see the following guardrail references:

• **Strongly Recommended Guardrails (p. 44)**
• **Elective Guardrails (p. 52)**

### Viewing Guardrail Details

In the guardrail details page of the console, you can find the following details for each guardrail:

• **Name** – The name of the guardrail.
• **Description** – A description of the guardrail.
• **Guidance** – The guidance is either mandatory, strongly recommended, or elective.
• **Category** – The guardrail category can be Audit Logs, Monitoring, Data Security, Network, IAM, or Control Tower Setup.
• **Behavior** – A guardrail's behavior is set to either preventive or detective.
• **Compliance Status** – A guardrail's compliance status can be clear, compliant, enforced, unknown, or in violation.

On the guardrail details page, you can also see guardrail artifacts. The guardrail is implemented by one or more artifacts. These artifacts can include a baseline AWS CloudFormation template, a service control policy to prevent account-level configuration changes or activity that may create configuration drift, and AWS Config Rules to detect account-level policy violations.

## Enabling Guardrails

Most guardrails are enabled automatically according to an OU's configuration, and some guardrails can be enabled manually on your OUs. The following procedure describes the steps for enabling guardrails on an OU.

**Important**

When you enable guardrails with strongly recommended guidance, AWS Control Tower managed AWS resources are created in your accounts. Do not modify or delete resources created by AWS Control Tower. Doing so could result in the guardrails entering an unknown state.

**To enable guardrails in an OU**

2. From the left navigation, choose **Guardrails**.
3. Choose a guardrail that you want to enable; for example, **Guardrail: Enable encryption for EBS volumes attached to EC2 instances**. This choice opens the guardrail's details page.
4. From **Organizational units enabled**, choose **Enable guardrail on OU**.
5. A new page is displayed that lists the names of your OUs. Identify the OU on which you want to enable this guardrail.
6. Choose **Enable guardrail on OU**.
7. Your guardrail is now enabled. It may take several minutes for the change to complete. When it does, you'll see that this guardrail is enabled on the OU you selected. You can enable only one guardrail at a time.

## Guardrail Reference

The following sections include a reference for each of the guardrails available in AWS Control Tower. Each guardrail reference includes the details, artifacts, additional information, and considerations to keep in mind when enabling a specific guardrail on a OU in your landing zone.

**Topics**

- Mandatory Guardrails (p. 32)
- Strongly Recommended Guardrails (p. 44)
- Elective Guardrails (p. 52)

## Mandatory Guardrails

Mandatory guardrails are enabled by default when you set up your landing zone and can't be disabled. Following, you'll find a reference for each of the mandatory guardrails available in AWS Control Tower.
Enable Encryption at Rest for Log Archive

This guardrail enables encryption at rest for the Amazon S3 buckets in the log archive account. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on the Core OU.

The artifact for this guardrail is the following service control policy (SCP).

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRAUDITBUCKETENCRYPTIONENABLED",
            "Effect": "Deny",
            "Action": [
                "s3:PutEncryptionConfiguration"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```

Enable Access Logging for Log Archive

This guardrail enables access logging in the log archive shared account. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on the Core OU.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRAUDITBUCKETLOGGINGENABLED",
            "Effect": "Deny",
            "Action": [
                "s3:PutBucketLogging"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```

Disallow Changes to CloudWatch Logs Log Groups

This guardrail prevents changes to CloudWatch Logs log groups that AWS Control Tower created in the log archive account when you set up your landing zone. It also prevents modifying retention policy in customer accounts. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on all OUs.

The artifact for this guardrail is the following SCP.
Disallow Deletion of AWS Config Aggregation Authorization

This guardrail prevents deletion of AWS Config aggregation authorizations that AWS Control Tower created in the audit account when you set up your landing zone. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRCONFIGAGGREGATIONAUTHORIZATIONPOLICY",
            "Effect": "Deny",
            "Action": [
                "config:DeleteAggregationAuthorization"
            ],
            "Resource": [
                "arn:aws:config:*:*:aggregation-authorization*
            ],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalArn": "arn:aws:iam::*:role/AWSControlTowerExecution"
                },
                "StringLike": {
                    "aws:ResourceTag/aws-control-tower": "managed-by-control-tower"
                }
            }
        }
    ]
}
```

Disallow Deletion of Log Archive

This guardrail prevents deletion of Amazon S3 buckets created by AWS Control Tower in the log archive account. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on the Core OU.
The artifact for this guardrail is the following SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GRAUDITBUCKETDELETIONPROHIBITED",
      "Effect": "Deny",
      "Action": [
        "s3:DeleteBucket"
      ],
      "Resource": [
        "arn:aws:s3:::aws-controltower*"
      ],
      "Condition": {
        "ArnNotLike": {
          "aws:PrincipalARN":"arn:aws:iam::*:role/AWSControlTowerExecution"
        }
      }
    }
  ]
}
```

**Disallow Policy Changes to Log Archive**

This guardrail disallows any policy changes from occurring in the log archive shared account. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on the Core OU.

The artifact for this guardrail is the following SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GRAUDITBUCKETPOLICYCHANGESPROHIBITED",
      "Effect": "Deny",
      "Action": [
        "s3:PutBucketPolicy"
      ],
      "Resource": ["*"],
      "Condition": {
        "ArnNotLike": {
          "aws:PrincipalARN":"arn:aws:iam::*:role/AWSControlTowerExecution"
        }
      }
    }
  ]
}
```

**Disallow Public Read Access to Log Archive**

This guardrail detects whether public read access is enabled to the Amazon S3 buckets in the log archive shared account. This guardrail does not change the status of the account. This is a detective guardrail with mandatory guidance. By default, this guardrail is enabled on the Core OU.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check that your S3 buckets do not allow public access
Parameters:
```
Disallow Public Write Access to Log Archive

This guardrail detects whether public write access is enabled to the Amazon S3 buckets in the log archive shared account. This guardrail does not change the status of the account. This is a detective guardrail with mandatory guidance. By default, this guardrail is enabled on the Core OU.

The artifact for this guardrail is the following AWS Config rule.

```
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check that your S3 buckets do not allow public access
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
Resources:
  CheckForS3PublicWrite:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks that your S3 buckets do not allow public write access. If an S3 bucket policy or bucket ACL allows public write access, the bucket is noncompliant.
      Source:
        Owner: AWS
        SourceIdentifier: S3_BUCKET_PUBLIC_WRITE_PROHIBITED
      Scope:
        ComplianceResourceTypes:
          - AWS::S3::Bucket
```

Set a Retention Policy for Log Archive

This guardrail sets a retention policy of 365 days on the logs in the log archive shared account. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on the Core OU.

The artifact for this guardrail is the following SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [ 
    {
      "Sid": "GRAUDITBUCKETRETENTIONPOLICY",
      "Effect": "Deny",
      "Action": [ "s3:PutLifecycleConfiguration"
    ],
```
Disallow Configuration Changes to CloudTrail

This guardrail prevents configuration changes to CloudTrail in your landing zone. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRCLOUDTRAILENABLED",
            "Effect": "Deny",
            "Action": [
                "cloudtrail:DeleteTrail",
                "cloudtrail:PutEventSelectors",
                "cloudtrail:StopLogging",
                "cloudtrail:UpdateTrail"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN":"arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```

Integrate CloudTrail Events with CloudWatch Logs

This guardrail performs real-time analysis of activity data by sending CloudTrail events to CloudWatch Logs log files. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled on all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRCLOUDTRAILENABLED",
            "Effect": "Deny",
            "Action": [
                "cloudtrail:DeleteTrail",
                "cloudtrail:PutEventSelectors",
                "cloudtrail:StopLogging",
                "cloudtrail:UpdateTrail"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN":"arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```
Enable CloudTrail in All Available Regions

This guardrail enables CloudTrail in all available AWS Regions. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRCLOUDTRAILENABLED",
            "Effect": "Deny",
            "Action": [
                "cloudtrail:DeleteTrail",
                "cloudtrail:PutEventSelectors",
                "cloudtrail:StopLogging",
                "cloudtrail:UpdateTrail"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN":"arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```

Enable Integrity Validation for CloudTrail Log File

This guardrail enables integrity validation for the CloudTrail log file in all accounts and OUs. It protects the integrity of account activity logs using CloudTrail log file validation, which creates a digitally signed digest file that contains a hash of each log that CloudTrail writes to Amazon S3. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRCLOUDTRAILENABLED",
            "Effect": "Deny",
            "Action": [
                "cloudtrail:DeleteTrail",
                "cloudtrail:PutEventSelectors",
                "cloudtrail:StopLogging",
                "cloudtrail:UpdateTrail"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN":"arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```
Disallow Changes to CloudWatch Set Up by AWS Control Tower

This guardrail disallows changes to CloudWatch as it was configured by AWS Control Tower when you set up your landing zone. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GRCLOUDWATCHEVENTPOLICY",
      "Effect": "Deny",
      "Action": [
        "events:PutRule",
        "events:PutTargets",
        "events:RemoveTargets",
        "events:DisableRule",
        "events:DeleteRule"
      ],
      "Resource": ["arn:aws:events:*:*:rule/aws-controltower-*"],
      "Condition": {
        "ArnNotLike": {
          "aws:PrincipalARN": "arn:aws:iam:*:role/AWSControlTowerExecution"
        }
      }
    }
  ]
}
```

Disallow Changes to AWS Config Aggregation Set Up by AWS Control Tower

This guardrail disallows changes to the AWS Config aggregation settings made by AWS Control Tower to collect configuration and compliance data when you set up your landing zone. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GRCONFIGRULETAGSPOLICY",
      "Effect": "Deny",
      "Action": [
        "config:TagResource",
        "config:UntagResource"
      ],
      "Resource": ["**"],
      "Condition": {
        "ArnNotLike": {
        }
      }
    }
  ]
}
```
Disallow Configuration Changes to AWS Config

This guardrail disallows configuration changes to AWS Config. It ensures that AWS Config records resource configurations in a consistent manner by disallowing AWS Config settings changes. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRCONFIGENABLED",
            "Effect": "Deny",
            "Action": [
                "config:DeleteConfigurationRecorder",
                "config:DeleteDeliveryChannel",
                "config:DeleteRetentionConfiguration",
                "config:PutConfigurationRecorder",
                "config:PutDeliveryChannel",
                "config:PutRetentionConfiguration",
                "config:StopConfigurationRecorder"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```

Enable AWS Config in All Available Regions

This guardrail enables AWS Config in all available AWS Regions. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRCONFIGENABLED",
            "Effect": "Deny",
            "Action": [
                "config:DeleteConfigurationRecorder",
                "config:DeleteDeliveryChannel",
                "config:DeleteRetentionConfiguration",
                "config:PutConfigurationRecorder",
                "config:PutDeliveryChannel"
            ],
            "Resource": ["*"],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```
Disallow Changes to AWS Config Rules Set Up by AWS Control Tower

This guardrail disallows changes to AWS Config Rules that were implemented by AWS Control Tower when the landing zone was set up. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
"Version": "2012-10-17",
"Statement": [ 
{
"Sid": "GRCONFIGRULEPOLICY",
"Effect": "Deny",
"Action": [ 
"config:PutConfigRule",
"config:DeleteConfigRule",
"config:DeleteEvaluationResults",
"config:DeleteConfigurationAggregator",
"config:PutConfigurationAggregator"
],
"Resource": ["*"],
"Condition": {
"ArnNotLike": {
"aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
},
"StringEquals": {
"aws:ResourceTag/aws-control-tower": "managed-by-control-tower"
}
}
]
}
```

Disallow Changes to IAM Roles Set Up by AWS Control Tower

This guardrail disallows changes to the IAM roles that were created by AWS Control Tower when the landing zone was set up. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
"Version": "2012-10-17",
"Statement": [ 
{
"Sid": "GRIAMROLEPOLICY",
"Effect": "Deny",
"Action": [ 
"iam:PassRole",
"iam:CreateRole"
],
"Resource": ["*"],
"Condition": {
"ArnNotLike": {
"aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
}
}
]
}
```
Disallow Changes to Lambda Functions Set Up by AWS Control Tower

This guardrail disallows changes to Lambda functions set up by AWS Control Tower. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRLAMBDAFUNCTIONPOLICY",
            "Effect": "Deny",
            "Action": [
                "lambda:AddPermission",
                "lambda:CreateEventSourceMapping",
                "lambda:CreateFunction",
                "lambda:DeleteEventSourceMapping",
                "lambda:DeleteFunction",
                "lambda:DeleteFunctionConcurrency",
                "lambda:PutFunctionConcurrency",
                "lambda:RemovePermission",
                "lambda:UpdateEventSourceMapping",
                "lambda:UpdateFunctionCode",
                "lambda:UpdateFunctionConfiguration"
            ],
            "Resource": [
                "arn:aws:lambda::*:*:function:aws-controltower-*"
            ],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN":"arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```
Disallow Changes to Amazon SNS Set Up by AWS Control Tower

This guardrail disallows changes to Amazon SNS set up by AWS Control Tower. It protects the integrity of Amazon SNS notification settings for your landing zone. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRSNSTOPICPOLICY",
            "Effect": "Deny",
            "Action": [
                "sns:AddPermission",
                "sns:CreateTopic",
                "sns:DeleteTopic",
                "sns:RemovePermission",
                "sns:SetTopicAttributes"
            ],
            "Resource": [
                "arn:aws:sns::*:*:aws-controltower-*"
            ],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```

Disallow Changes to Amazon SNS Subscriptions Set Up by AWS Control Tower

This guardrail disallows changes to Amazon SNS subscriptions set up by AWS Control Tower. It protects the integrity of Amazon SNS subscriptions settings for your landing zone. This is a preventive guardrail with mandatory guidance. By default, this guardrail is enabled in all OUs.

The artifact for this guardrail is the following SCP.

```json
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "GRSNSSUBSCRIPTIONPOLICY",
            "Effect": "Deny",
            "Action": [
                "sns:Subscribe",
                "sns:Unsubscribe"
            ],
            "Resource": [
                "arn:aws:sns::*:*:aws-controltower-SecurityNotifications"
            ],
            "Condition": {
                "ArnNotLike": {
                    "aws:PrincipalARN": "arn:aws:iam::*:role/AWSControlTowerExecution"
                }
            }
        }
    ]
}
```
Strongly Recommended Guardrails

Strongly recommended guardrails are based on best practices for well-architected multi-account environments. These guardrails are not enabled by default, and can be disabled. Following, you'll find a reference for each of the strongly recommended guardrails available in AWS Control Tower.

Disallow Creation of Access Keys for the Root User

Secures your AWS accounts by disallowing creation of access keys for the root user. We recommend that you instead create access keys for the IAM users with limited permissions to interact with your AWS account. This is a preventive guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following SCP.

```json
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "GRRESTRICTROOTUSERACCESSKEYS",
         "Effect": "Deny",
         "Action": "iam:CreateAccessKey",
         "Resource": ["*"]
      },
      {
         "Condition": {
            "StringLike": {
               "aws:PrincipalArn": [
                  "arn:aws:iam::*:root"
               ]
            }
         }
      }
   ]
}
```

Disallow Actions as a Root User

Secures your AWS accounts by disallowing account access with root user credentials, which are credentials of the account owner that allow unrestricted access to all resources in the account. Instead, we recommend that you create AWS Identity and Access Management (IAM) users for everyday interaction with your AWS account. This is a preventive guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following SCP.

```json
{
   "Version": "2012-10-17",
   "Statement": [
      {
         "Sid": "GRRESTRICTROOTUSER",
         "Effect": "Deny",
         "Action": "*",
         "Resource": [
          "*
        ]
      }
   ]
}
```
Enable Encryption for Amazon EBS Volumes Attached to Amazon EC2 Instances

This guardrail detects whether encryption is enabled for Amazon EBS volumes attached to Amazon EC2 instances in your landing zone. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail isn't enabled on any OUs.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check for encryption of all storage volumes attached to compute
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
Resources:
  CheckForEncryptedVolumes:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks whether EBS volumes that are in an attached state are encrypted.
      Source:
        Owner: AWS
        SourceIdentifier: ENCRYPTED_VOLUMES
      Scope:
        ComplianceResourceTypes:
        - AWS::EC2::Volume
```

Disallow Internet Connection Through RDP

This guardrail detects whether internet connections are enabled to Amazon EC2 instances through services like Remote Desktop Protocol (RDP). This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check whether security groups that are in use disallow unrestricted incoming TCP traffic to the specified ports.
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
blockedPort1:
```
Type: String  
Default: '20'  
Description: Blocked TCP port number.

blockedPort2:  
Type: String  
Default: '21'  
Description: Blocked TCP port number.

blockedPort3:  
Type: String  
Default: '3389'  
Description: Blocked TCP port number.

blockedPort4:  
Type: String  
Default: '3306'  
Description: Blocked TCP port number.

blockedPort5:  
Type: String  
Default: '4333'  
Description: Blocked TCP port number.

Conditions:  
blockedPort1:  
  Fn::Not:  
    - Fn::Equals:  
      - ''  
      - Ref: blockedPort1

blockedPort2:  
  Fn::Not:  
    - Fn::Equals:  
      - ''  
      - Ref: blockedPort2

blockedPort3:  
  Fn::Not:  
    - Fn::Equals:  
      - ''  
      - Ref: blockedPort3

blockedPort4:  
  Fn::Not:  
    - Fn::Equals:  
      - ''  
      - Ref: blockedPort4

blockedPort5:  
  Fn::Not:  
    - Fn::Equals:  
      - ''  
      - Ref: blockedPort5

Resources:  
CheckForRestrictedCommonPortsPolicy:  
  Type: AWS::Config::ConfigRule  
Properties:  
    ConfigRuleName: !Sub ${ConfigRuleName}
    Description: Checks whether security groups that are in use disallow unrestricted incoming TCP traffic to the specified ports.
    InputParameters:  
      blockedPort1:  
        Fn::If:  
          - blockedPort1  
          - Ref: blockedPort1  
          - Ref: AWS::NoValue

      blockedPort2:  
        Fn::If:  
          - blockedPort2  
          - Ref: blockedPort2  
          - Ref: AWS::NoValue

      blockedPort3:  
        Fn::If:  
          - blockedPort3

46
Disallow Internet Connection Through SSH

This guardrail detects whether any internet connections are allowed through remote services like the Secure Shell (SSH) protocol. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check whether security groups that are in use disallow SSH
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
Resources:
  CheckForRestrictedSshPolicy:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks whether security groups that are in use disallow unrestricted incoming SSH traffic.
      Scope:
        ComplianceResourceTypes:
        - AWS::EC2::SecurityGroup
      Source:
        Owner: AWS
        SourceIdentifier: INCOMING_SSH_DISABLED
```

Enable MFA for the Root User

This guardrail detects whether multi-factor authentication (MFA) is enabled for the root user of the master account. MFA reduces vulnerability risks from weak authentication by adding an extra authentication code on top of a user name and password. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to require MFA for root access to accounts
Parameters:
```
Disallow Public Read Access to Amazon S3 Buckets

This guardrail detects whether public read access is allowed to Amazon S3 buckets. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check that your S3 buckets do not allow public access
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
Resources:
  CheckForS3PublicRead:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks that your S3 buckets do not allow public read access. If an S3 bucket policy or bucket ACL allows public read access, the bucket is noncompliant.
      Source:
        Owner: AWS
        SourceIdentifier: S3_BUCKET_PUBLIC_READ_PROHIBITED
```

Disallow Public Write Access to Amazon S3 Buckets

This guardrail detects whether public write access is allowed to Amazon S3 buckets. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check that your S3 buckets do not allow public access
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
Resources:
  CheckForS3PublicWrite:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks that your S3 buckets do not allow public write access. If an S3 bucket policy or bucket ACL allows public write access, the bucket is noncompliant.
      Source:
        Owner: AWS
        SourceIdentifier: S3_BUCKET_PUBLIC_WRITE_PROHIBITED
      Scope:
        ComplianceResourceTypes:
          - AWS::S3::Bucket
```

Disallow Amazon EBS Volumes That Are Unattached to An Amazon EC2 Instance

Detects whether an Amazon EBS volume persists independently from an Amazon EC2 instance. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check whether EBS volumes are attached to EC2 instances
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
  deleteOnTermination:
    Type: 'String'
    Default: 'None'
    Description: 'Check for Delete on termination'
Conditions:
  deleteOnTermination:
    Fn::Not:
      - Fn::Equals:
        - 'None'
        - Ref: deleteOnTermination
Resources:
  CheckForEc2VolumesInUse:
```
Disallow Amazon EC2Instance Types That Are Not Amazon EBS-Optimized

Detects whether Amazon EC2 instances are launched without an Amazon EBS volume that is performance optimized. Amazon EBS-optimized volumes minimize contention between Amazon EBS I/O and other traffic from your instance. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

Disallow Public Access to Amazon RDS Database Instances

Detects whether your Amazon RDS database instances have public access enabled. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.
Disallow Public Access to Amazon RDS Database Snapshots

Detects whether your Amazon RDS database snapshots have public access enabled. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```json
AWSTemplateFormatVersion: 2010-09-09
Description: Checks if Amazon Relational Database Service (Amazon RDS) snapshots are public.
Parameters:
ConfigRuleName:
  Type: 'String'
  Description: 'Name for the Config rule'
Resources:
CheckForRdsStorageEncryption:
  Type: AWS::Config::ConfigRule
  Properties:
    ConfigRuleName: !Sub ${ConfigRuleName}
    Description: Checks if Amazon Relational Database Service (Amazon RDS) snapshots are public. The rule is non-compliant if any existing and new Amazon RDS snapshots are public.
    Source:
      Owner: AWS
      SourceIdentifier: RDS_SNAPSHOTS_PUBLIC_PROHIBITED
    Scope:
      ComplianceResourceTypes:
        - AWS::RDS::DBSnapshot
```

Disallow Amazon RDS Database Instances That Are Not Storage Encrypted

Detects whether your Amazon RDS database instances are not encrypted at rest, along with their automated backups, Read Replicas, and snapshots. This guardrail does not change the status of the account. This is a detective guardrail with strongly recommended guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```json
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check whether storage encryption is enabled for your RDS DB instances
```
Parameters:
ConfigRuleName:
  Type: 'String'
  Description: 'Name for the Config rule'

Resources:
CheckForRdsStorageEncryption:
  Type: AWS::Config::ConfigRule
  Properties:
    ConfigRuleName: !Sub ${ConfigRuleName}
    Description: Checks whether storage encryption is enabled for your RDS DB instances.
    Source:
      Owner: AWS
      SourceIdentifier: RDS_STORAGE_ENCRYPTED
    Scope:
      ComplianceResourceTypes:
        - AWS::RDS::DBInstance

Elective Guardrails

Elective guardrails enable you to lock down or track attempts at performing commonly restricted actions in an AWS enterprise environment. These guardrails are not enabled by default, and can be disabled. Following, you'll find a reference for each of the elective guardrails available in AWS Control Tower.

Disallow Cross-Region Replication for Amazon S3 Buckets

Restricts the location of your Amazon S3 data to a single AWS Region by disabling any automatic, asynchronous copying of objects across buckets to other AWS Regions. This is a preventive guardrail with elective guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GRRESTRICTS3CROSSREGIONREPLICATION",
      "Effect": "Deny",
      "Action": [
        "s3:PutReplicationConfiguration"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

Disallow Delete Actions on Amazon S3 Buckets Without MFA

Protects your Amazon S3 buckets by requiring MFA for delete actions. MFA adds an extra authentication code on top of a user name and password. This is a preventive guardrail with elective guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following SCP.

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "GRRESTRICTS3DELETEWITHOUTMFA",
      "Effect": "Deny",
      "Action": [
        "s3:DeleteObject"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```
Disallow Access to IAM Users Without MFA

Protects your account by requiring MFA for all IAM users in the account. MFA adds an extra authentication code on top of a username and password. This guardrail detects whether MFA is enabled. This guardrail does not change the status of the account. This is a detective guardrail with elective guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check whether the IAM users have MFA enabled
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
  MaximumExecutionFrequency:
    Type: String
    Default: 1hour
    Description: The frequency that you want AWS Config to run evaluations for the rule.
    AllowedValues:
      - 1hour
      - 3hours
      - 6hours
      - 12hours
      - 24hours
Mappings:
  Settings:
    FrequencyMap:
      1hour : One_Hour
      3hours : Three_Hours
      6hours : Six_Hours
      12hours : Twelve_Hours
      24hours : TwentyFour_Hours
Resources:
  CheckForIAMUserMFA:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks whether the AWS Identity and Access Management users have multi-factor authentication (MFA) enabled. The rule is COMPLIANT if MFA is enabled.
      Source:
        Owner: AWS
        SourceIdentifier: IAM_USER_MFA_ENABLED
      MaximumExecutionFrequency:
```
Disallow Console Access to IAM Users Without MFA

Protects your account by requiring MFA for all IAM users in the console. MFA adds an extra authentication code on top of a username and password. This guardrail detects whether MFA is enabled. This guardrail does not change the status of the account. This is a detective guardrail with elective guidance. By default, this guardrail is not enabled.

The artifact for this guardrail is the following AWS Config rule.

```yaml
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check whether MFA is enabled for all AWS IAM users that use a console password.
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
  MaximumExecutionFrequency:
    Type: String
    Default: 1hour
    Description: The frequency that you want AWS Config to run evaluations for the rule.
    AllowedValues:
      - 1hour
      - 3hours
      - 6hours
      - 12hours
      - 24hours
Mappings:
  Settings:
    FrequencyMap:
      1hour   : One_Hour
      3hours  : Three_Hours
      6hours  : Six_Hours
      12hours : Twelve_Hours
      24hours : TwentyFour_Hours
Resources:
  CheckForIAMUserConsoleMFA:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks whether AWS Multi-Factor Authentication (MFA) is enabled for all AWS Identity and Access Management (IAM) users that use a console password. The rule is COMPLIANT if MFA is enabled.
      Source:
        Owner: AWS
        SourceIdentifier: MFA_ENABLED_FOR_IAM_CONSOLE_ACCESS
      MaximumExecutionFrequency:
        !FindInMap
        - Settings
        - FrequencyMap
        - !Ref MaximumExecutionFrequency
```

Disallow Amazon S3 Buckets That Are Not Versioning Enabled

Detects whether your Amazon S3 buckets are not versioning enabled. Versioning allows you to recover objects from accidental deletion or overwrite. This guardrail does not change the status of the account. This is a detective guardrail with elective guidance. By default, this guardrail is not enabled.
The artifact for this guardrail is the following AWS Config rule.

```
AWSTemplateFormatVersion: 2010-09-09
Description: Configure AWS Config rules to check whether versioning is enabled for your S3 buckets.
Parameters:
  ConfigRuleName:
    Type: 'String'
    Description: 'Name for the Config rule'
Resources:
  CheckForS3VersioningEnabled:
    Type: AWS::Config::ConfigRule
    Properties:
      ConfigRuleName: !Sub ${ConfigRuleName}
      Description: Checks whether versioning is enabled for your S3 buckets.
      Source:
        Owner: AWS
        SourceIdentifier: S3_BUCKET_VERSIONING_ENABLED
      Scope:
        ComplianceResourceTypes:
        - AWS::S3::Bucket
```
Integrated Services

AWS Control Tower is a well-architected service that's built on top of other AWS services. This chapter provides a brief overview of these services, including configuration information about the following services and how they work in AWS Control Tower.

Topics
- Scripting Environments with AWS CloudFormation (p. 56)
- Monitoring Events with CloudTrail (p. 56)
- Monitoring Resources and Services with CloudWatch (p. 56)
- Govern Resource Configurations with AWS Config (p. 57)
- Manage Permissions for Entities with IAM (p. 57)
- Run Serverless Compute Functions with Lambda (p. 57)
- Manage Accounts Through AWS Organizations (p. 57)
- Store Objects with Amazon S3 (p. 58)
- Provisioning Accounts Through AWS Service Catalog (p. 58)
- Managing Users and Access Through AWS Single Sign-On (p. 58)
- Tracking Alerts Through Amazon Simple Notification Service (p. 62)
- Build Distributed Applications with AWS Step Functions (p. 62)

Scripting Environments with AWS CloudFormation

AWS CloudFormation enables you to create and provision AWS infrastructure deployments predictably and repeatedly. It helps you leverage AWS products to build highly reliable, highly scalable, cost-effective applications in the cloud without worrying about creating and configuring the underlying AWS infrastructure. AWS CloudFormation enables you to use a template file to create and delete a collection of resources together as a single unit (a stack). For more information, see AWS CloudFormation User Guide.

AWS Control Tower uses AWS CloudFormation stacksets to apply guardrails on accounts.

Monitoring Events with CloudTrail

With AWS CloudTrail, you can monitor your AWS deployments in the cloud by getting a history of AWS API calls for your accounts. You can also identify which users and accounts called AWS APIs for services that support CloudTrail, the source IP address the calls were made from, and when the calls occurred. You can integrate CloudTrail into applications using the API, automate trail creation for your organization, check the status of your trails, and control how administrators turn CloudTrail logging on and off. For more information, see AWS CloudTrail User Guide.

Monitoring Resources and Services with CloudWatch

Amazon CloudWatch provides a reliable, scalable, and flexible monitoring solution that you can start using within minutes. You no longer need to set up, manage, and scale your own monitoring systems and infrastructure. For more information, see Amazon CloudWatch User Guide.
Govern Resource Configurations with AWS Config

AWS Config provides a detailed view of the resources associated with your AWS account, including how they are configured, how they are related to one another, and how the configurations and their relationships have changed over time. For more information, see AWS Config Developer Guide.

AWS Config resources provisioned by AWS Control Tower are tagged automatically with aws-control-tower and a value of managed-by-control-tower.

AWS Control Tower uses AWS Config Rules with some guardrails. For more information, see Guardrails in AWS Control Tower (p. 30).

Manage Permissions for Entities with IAM

AWS Identity and Access Management (IAM) is a web service for securely controlling access to AWS services. With IAM, you can centrally manage users, security credentials such as access keys, and permissions that control which AWS resources users and applications can access.

When you set up your landing zone, a number of groups are created for AWS SSO. These groups have permission sets that are pre-defined permissions policies from IAM. Your end users can also use IAM to define the scope of permissions for IAM users and other entities within member accounts.

Run Serverless Compute Functions with Lambda

With AWS Lambda, you can run code without provisioning or managing servers. You can run code for virtually any type of application or backend service—all with zero administration. Just upload your code and Lambda takes care of everything required to run and scale your code with high availability. You can set up your code to automatically trigger from other AWS services or call it directly from any web or mobile app.

Manage Accounts Through AWS Organizations

AWS Organizations is an account management service that lets you consolidate multiple AWS accounts into an organization that you create and centrally manage. With Organizations, you can create member accounts and invite existing accounts to join your organization. You can organize those accounts into groups and attach policy-based controls. For more information, see AWS Organizations User Guide.

In AWS Control Tower, Organizations helps centrally manage billing; control access, compliance, and security; and share resources across your member AWS accounts. Accounts are grouped into logical groups, called organizational units (OUs). For more information on Organizations, see AWS Organizations User Guide.

AWS Control Tower uses the following OUs:

- **Root** – The parent container for all accounts and all other OUs in your landing zone.
- **Core** – This OU contains the log archive account, the audit account, and the resources they own.
- **Custom OU** – This OU is created when you set up your landing zone. It and other child OUs in your landing zone contain your member accounts. These are the accounts that your end users access to perform work on AWS resources.
Considerations

You can add additional OUs in your landing zone through the AWS Control Tower console on the Organizational units page.

OUs created through AWS Control Tower can have guardrails applied to them. OUs created outside of AWS Control Tower cannot, and they are not displayed in AWS Control Tower.

Store Objects with Amazon S3

Amazon Simple Storage Service (Amazon S3) is storage for the internet. You can use Amazon S3 to store and retrieve any amount of data at any time, from anywhere on the web. You can accomplish these tasks using the simple and intuitive web interface of the AWS Management Console. For more information, see Amazon Simple Storage Service Console User Guide.

When you set up your landing zone, an Amazon S3 bucket is created in your log archive account to contain all logs across all accounts in your landing zone.

Provisioning Accounts Through AWS Service Catalog

AWS Service Catalog enables IT administrators to create, manage, and distribute portfolios of approved products to end users, who can then access the products they need in a personalized portal. Typical products include servers, databases, websites, or applications that are deployed using AWS resources. You can control which users have access to specific products to enforce compliance with organizational business standards, manage product lifecycles, and help users find and launch products with confidence. For more information, see AWS Service Catalog Administrator Guide.

In AWS Control Tower, your central cloud administrators and your end users can provision accounts in your landing zone using Account Factory, a product in AWS Service Catalog. For more information, see Account Factory (p. 63).

Managing Users and Access Through AWS Single Sign-On

AWS Single Sign-On is a cloud-based service that simplifies how you manage SSO access to AWS accounts and business applications. You can control SSO access and user permissions across all your AWS accounts in AWS Organizations. You also can administer access to popular business applications and custom applications that support Security Assertion Markup Language (SAML) 2.0. Also, AWS SSO offers a user portal where your users can find all their assigned AWS accounts, business applications, and custom applications in one place. For more information, see AWS Single Sign-On User Guide.

Working With AWS SSO and AWS Control Tower

In AWS Control Tower, AWS Single Sign-On allows central cloud administrators and end users to manage access to multiple AWS accounts and business applications. AWS Control Tower uses this service to set up and manage access to the accounts created through AWS Service Catalog.
For a brief tutorial about how to set up your SSO users and permissions in AWS Control Tower, you can view this video (6:23). For better viewing, select the icon at the lower right corner of the video to enlarge it to full screen. Captioning is available.

**Video Walkthrough of Setting Up AWS SSO in AWS Control Tower.**

When you initially set up AWS Control Tower, only the root user and any IAM users with the correct permissions can add AWS SSO users. However, after end users have been added in the AWSAccountFactory group, they can create new SSO users from the Account Factory wizard. For more information, see Account Factory (p. 63).

Your landing zone is set up with a preconfigured directory that helps you manage user identities and single sign-on, so that your users have federated access across accounts. When you set up your landing zone, this default directory is created to contain *user groups* and *permission sets*.

**User Groups, Roles, and Permission Sets**

User groups manage specialized *roles* that are defined within your shared accounts. Roles establish sets of permissions that belong together. All members of a group inherit the permission sets, or roles, associated with the group. You can create new groups for the end users of your member accounts, so that you can custom-assign only the roles that are needed for the specific tasks a group performs.

The permission sets available cover a broad range of distinct user permission requirements, such as read-only access, AWS Control Tower administrative access, and AWS Service Catalog access. These permission sets enable your end users to provision their own AWS accounts in your landing zone quickly, and in compliance with your enterprise's guidelines.

For tips on planning your allocations of users, groups, and permissions, refer to Recommendations for Setting Up Groups, Roles, and Policies (p. 23)

For more information on how to use this service in the context of AWS Control Tower, see the following topics in the AWS Single Sign-On User Guide.

- To add users, see Add Users.
- To add users to groups, see Add Users to Groups.
- To edit user properties, see Edit User Properties.
- To add a group, see Add Groups.

**Warning**

AWS Control Tower sets up your AWS SSO directory in your home region. If you set up your landing zone in another Region and then navigate to the AWS SSO console, you must change the Region to your home region. Do not delete your AWS SSO configuration in your home region.

**Things to Know About SSO Accounts and AWS Control Tower**

Here are some good things to know when working with AWS SSO user accounts in AWS Control Tower.

- If your AWS SSO user account is disabled, you'll get an error message when trying to provision new accounts in Account Factory. You can re-enable your SSO user in the AWS SSO console.
- If you specify a new SSO user email address when you update the provisioned product associated with an account that was vended by Account Factory, AWS Control Tower creates a new SSO user account. The previously created user account is not removed. If you prefer to remove the previous SSO user email address from AWS SSO, see Disabling a User.
- AWS SSO has been integrated with Azure Active Directory, and you can connect your existing Azure Active Directory to AWS Control Tower. Learn more in this blog post.
- For more information about how the behavior of AWS Control Tower interacts with AWS SSO and different identity sources, refer to the Considerations for Changing Your Identity Source in the AWS SSO documentation.

**AWS SSO Groups for AWS Control Tower**

AWS Control Tower offers preconfigured groups to organize users that perform specific tasks in your accounts. You can add users and assign them to these groups directly in AWS SSO. Doing so matches permission sets to users in groups within your accounts. The groups created when you set up your landing zone are as follows.

**AWSAccountFactory**

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master account</td>
<td>AWSServiceCatalogEndUserAccess</td>
<td>This group is only used in this account to provision new accounts using Account Factory.</td>
</tr>
</tbody>
</table>

**AWSServiceCatalogAdmins**

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master account</td>
<td>AWSServiceCatalogAdminFullAccess</td>
<td>This group is only used in this account to make administrative changes to Account Factory. Users in this group can't provision new accounts unless they're also in the AWSAccountFactory group.</td>
</tr>
</tbody>
</table>

**AWSControlTowerAdmins**

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master account</td>
<td>AWSAdministratorAccess</td>
<td>Users of this group in this account are the only ones that have access to the AWS Control Tower console.</td>
</tr>
<tr>
<td>Log archive account</td>
<td>AWSAdministratorAccess</td>
<td>Users have administrator access in this account.</td>
</tr>
<tr>
<td>Audit account</td>
<td>AWSAdministratorAccess</td>
<td>Users have administrator access in this account.</td>
</tr>
<tr>
<td>Member accounts</td>
<td>AWSOrganizationsFullAccess</td>
<td>Users have full access to Organizations in this account.</td>
</tr>
</tbody>
</table>

**AWSSecurityAuditPowerUsers**

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master account</td>
<td>AWSPowerUserAccess</td>
<td>Users can perform application development tasks and can</td>
</tr>
<tr>
<td>Account</td>
<td>Permission sets</td>
<td>Description</td>
</tr>
<tr>
<td>------------------</td>
<td>---------------------------</td>
<td>------------------------------------------------------------------------------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>create and configure resources and services that support AWS aware application development.</td>
</tr>
<tr>
<td>Log archive account</td>
<td>AWSPowerUserAccess</td>
<td>Users can perform application development tasks and can create and configure resources and services that support AWS aware application development.</td>
</tr>
<tr>
<td>Audit account</td>
<td>AWSPowerUserAccess</td>
<td>Users can perform application development tasks and can create and configure resources and services that support AWS aware application development.</td>
</tr>
<tr>
<td>Member accounts</td>
<td>AWSPowerUserAccess</td>
<td>Users can perform application development tasks and can create and configure resources and services that support AWS aware application development.</td>
</tr>
</tbody>
</table>

**AWSSecurityAuditors**

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Master account</td>
<td>AWSReadOnlyAccess</td>
<td>Users have read-only access to all AWS services and resources in this account.</td>
</tr>
<tr>
<td>Log archive account</td>
<td>AWSReadOnlyAccess</td>
<td>Users have read-only access to all AWS services and resources in this account.</td>
</tr>
<tr>
<td>Audit account</td>
<td>AWSReadOnlyAccess</td>
<td>Users have read-only access to all AWS services and resources in this account.</td>
</tr>
<tr>
<td>Member accounts</td>
<td>AWSReadOnlyAccess</td>
<td>Users have read-only access to all AWS services and resources in this account.</td>
</tr>
</tbody>
</table>

**AWSLogArchiveAdmins**

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log archive account</td>
<td>AWSAdministratorAccess</td>
<td>Users have administrator access in this account.</td>
</tr>
</tbody>
</table>
AWSLogArchiveViewers

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log archive account</td>
<td>AWSReadOnlyAccess</td>
<td>Users have read-only access to all AWS services and resources in this account.</td>
</tr>
</tbody>
</table>

AWSAuditAccountAdmins

<table>
<thead>
<tr>
<th>Account</th>
<th>Permission sets</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Audit account</td>
<td>AWSAdministratorAccess</td>
<td>Users have administrator access in this account.</td>
</tr>
</tbody>
</table>

Tracking Alerts Through Amazon Simple Notification Service

Amazon Simple Notification Service (Amazon SNS) is a web service that enables applications, end-users, and devices to send and receive notifications instantly from the cloud. For more information, see *Amazon Simple Notification Service Developer Guide*.

AWS Control Tower uses Amazon SNS to send programmatic alerts to the email addresses of your master account and your audit account. These alerts help you prevent drift within your landing zone. For more information, see *Detecting and Resolving Drift in AWS Control Tower* (p. 71).

Build Distributed Applications with AWS Step Functions

AWS Step Functions makes it easy to coordinate the components of distributed applications as a series of steps in a visual workflow. You can quickly build and run state machines to execute the steps of your application in a reliable and scalable fashion. For more information, see *AWS Step Functions Developer Guide*.
Account Factory

This chapter includes an overview and procedures for provisioning new accounts in your AWS Control Tower landing zone. AWS Control Tower provides three methods for creating member accounts:

- through the Account Factory console that is part of AWS Service Catalog.
- through the Enroll account feature within AWS Control Tower.
- from your AWS Control Tower landing zone's master account, using Lambda code and appropriate IAM roles.

The standard way to provision accounts is through Account Factory, a console-based product that's part of the AWS Service Catalog. If your landing zone is not in a state of drift, you can use Enroll account. Also, some customers may prefer to configure new accounts programmatically using IAM roles and Lambda functions.

With the appropriate user group permissions, provisioners can specify standardized baselines and network configurations for all accounts in your organization.

Permissions for Configuring and Provisioning Accounts

The AWS Control Tower account factory enables cloud administrators and AWS Single Sign-On end users to provision accounts in your landing zone. By default, AWS SSO users that provision accounts must be in the AWSAccountFactory group or the master group.

Note
Exercise caution when working from the master account, as you would when using any account that has generous permissions across your organization.

The AWS Control Tower master account has a trust relationship with the AWSControlTowerExecution role, which enables account setup from the master account, including some automated account setup. For more information about the AWSControlTowerExecution role, see How AWS Control Tower Works With Roles to Create and Manage Accounts (p. 9).

To enroll an existing AWS account into AWS Control Tower, that account must have the AWSControlTowerExecution role enabled. For more information about how to enroll an existing account, see Enrolling an Existing AWS Account in AWS Control Tower (p. 114).

Automated Account Provisioning With IAM Roles

To configure Account Factory accounts in a more automated way, you can create Lambda functions in the AWS Control Tower master account, which assumes the AWSControlTowerExecution role in the member account. Then, using the role, the master account performs the desired configuration steps in each member account.
If you're provisioning accounts using Lambda functions, the identity that will perform this work must have the following IAM permissions policy, in addition to `AWSServiceCatalogEndUserFullAccess`.

```json
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "AWSControlTowerAccountFactoryAccess",
      "Effect": "Allow",
      "Action": [
        "sso:GetProfile",
        "sso:CreateProfile",
        "sso:UpdateProfile",
        "sso:AssociateProfile",
        "sso:CreateApplicationInstance",
        "sso:GetSSOS Status",
        "sso:GetTrust",
        "sso:CreateTrust",
        "sso:UpdateTrust",
        "sso:GetPeregrineStatus",
        "sso:GetApplicationInstance",
        "sso:ListDirectoryAssociations",
        "sso:ListPermissionSets",
        "sso:GetPermissionSet",
        "sso:ProvisionApplicationInstanceForAWSAccount",
        "sso:ProvisionApplicationProfileForAWSAccountInstance",
        "sso:ProvisionSAMLProvider",
        "sso:ListProfileAssociations",
        "sso-directory:ListMembersInGroup",
        "sso-directory:AddMemberToGroup",
        "sso-directory:SearchGroups",
        "sso-directory:SearchGroupsWithGroupName",
        "sso-directory:SearchUsers",
        "sso-directory:CreateUser",
        "sso-directory:DescribeGroups",
        "sso-directory:DescribeDirectory",
        "sso-directory:GetUserPoolInfo",
        "controltower:CreateManagedAccount",
        "controltower:DescribeManagedAccount",
        "controltower:DeregisterManagedAccount",
        "s3:GetObject",
        "organizations:describeOrganization",
        "sso:DescribeRegisteredRegions"
      ],
      "Resource": "*"
    }
  ]
}
```

Create or Enroll An Individual Account

The **Enroll account** feature is available in AWS Control Tower for provisioning new accounts in your landing zone and for enrolling existing AWS accounts so that they are governed by AWS Control Tower.

The **Enroll account** capability is available when your landing zone is not in a state of drift. To view this capability:

- Navigate to the **Account Factory** page in AWS Control Tower.
- Select the **Enroll account** item near the top of the page.
- You'll then see a **Create account** section, where you can fill in the required fields: *account email*, *account name*, *SSO user name*, and *organizational unit*.
When you've filled in the information, select **Enroll account**.

You'll see a flashbar confirming that your account enrollment process has been successfully submitted. If an error has occurred, AWS Control Tower may ask you for corrections. The account provisioning process may take several minutes.

**Note**
If you are enrolling an existing AWS account, be sure to type the existing email address correctly. Otherwise, a new account will be created.

Certain errors may require that you refresh the page and try again. If your landing zone is in a state of drift, you may not be able to use the **Enroll account** capability successfully. You'll need to provision new accounts through AWS Service Catalog until your landing zone drift has been resolved.

When you enroll accounts, you must be signed into an account with an IAM user that has the `AWSServiceCatalogEndUserFullAccess` policy enabled, and you cannot be signed in as **Root**.

Accounts that you enroll must be updated by means of the AWS Service Catalog and the AWS Control Tower account factory, as you would update any other account. Update procedures are given in the section called **Updating and Moving Account Factory Accounts (p. 66)**.

## Provisioning Account Factory Accounts With AWS Service Catalog

The following procedure describes how to provision accounts as an AWS SSO end user, through AWS Service Catalog. This procedure also is referred to as **advanced account provisioning**.

**To provision accounts in Account Factory as an end user**

1. Sign in from your user portal URL.
2. From **Your applications**, choose **AWS Account**.
3. From the list of accounts, choose the account ID for your master account. This ID may also have a label, for example, **(Master)**.
4. Ensure that you've selected the correct AWS Region for provisioning accounts, which should be your AWS Control Tower home region.
5. From `AWSServiceCatalogEndUserAccess`, choose **Management console**. This opens the AWS Management Console for this user in this account.
6. Search for and choose **Service Catalog** to open the AWS Service Catalog console.
7. From the navigation pane, choose **Products list**.
8. Select **AWS Control Tower Account Factory**, then choose the **Launch** button. This selection starts the wizard to provision a new account.
9. Fill in the information, and keep the following in mind:
   - The **SSOUserEmail** can be a new email address, or the email address associated with an existing AWS SSO user. Whichever you choose, this user will have administrative access to the account you're provisioning.
   - The **AccountEmail** must be an email address that isn't already associated with an AWS account. If you used a new email address in **SSOUserEmail**, you can use that email address here.
10. When you’re finished, choose **Next** until you get to the **Review** page of the wizard. Do not define **TagOptions** and do not enable **Notifications**, otherwise the account can fail to be provisioned.
11. Review your account settings, and then choose **Launch**. Do not create a resource plan, otherwise the account will fail to be provisioned.

12. Your account is now being provisioned. It can take a few minutes to complete. You can refresh the page to update the displayed status information. Only one account can be provisioned at a time.

**Tips on Managing Account Factory Accounts**

Accounts that you provision through the AWS Control Tower Account Factory can be updated, they can be closed, or they can be repurposed. For example, you can repurpose existing accounts for other workloads and other users by updating the email addresses and user parameters for the account.

If you specify a new SSO user email address when you update the provisioned product associated with an account that was vended by account factory, AWS Control Tower creates a new SSO user account. The previously created user account is not removed. If you prefer to remove the previous SSO user email from AWS SSO, see **Disabling a User**.

With Account Factory you also can change the organizational unit (OU) for an account, or you can unmanage an account, by following the procedures in this chapter. For more information on unmanaging an account, see **Unmanaging a Member Account** (p. 68). Certain updates require that you or an administrator must **Sign in as a Root User** (p. 23) to the account, to gain appropriate permissions.

**Updating and Moving Account Factory Accounts**

The following procedure guides you through how to update your Account Factory account or move it to a new OU.

**To update an Account Factory account or change its OU**

   
   **Note**
   
   You must be signed in as a user with the permissions to provision new products in AWS Service Catalog; for example, an AWS SSO user in either the `AWSAccountFactory` or `AWSServiceCatalogAdmins` groups.

2. Choose **Provision new account** to open the AWS Service Catalog console and the Account Factory product.

3. From the navigation pane, choose **Provisioned products list**.

4. For each account listed, perform the following steps to update all your member accounts:
   a. From the drop-down menu for the account, choose **Provisioned product details**.
   b. Make a note of the following parameters:
      - **SSOUserEmail** (Available in provisioned product details)
      - **AccountEmail** (Available in provisioned product details)
      - **SSOUserFirstName** (Available in SSO)
      - **SSOUserLastName** (Available in SSO)
      - **AccountName** (Available in SSO)
   c. From **Actions**, choose **Update**.
   d. Choose the button next to the **Version** of the product you want to update, and choose **Next**.
   e. Provide the parameter values that were mentioned previously.
• If you want to keep the existing OU, for ManagedOrganizationalUnit, choose the OU that the account was already in.
• If you want to migrate the account to a new OU, for ManagedOrganizationalUnit, choose the new OU for the account.

A central cloud administrator can find this information in the AWS Control Tower console, under Accounts.
f. Choose Next.
g. Review your changes, and then choose Update. This process can take a few minutes per account.

Configuring Account Factory with Amazon Virtual Private Cloud Settings

Account Factory enables you to create pre-approved baselines and configuration options for accounts in your organization. You can configure and provision new accounts through AWS Service Catalog.

On the Account Factory page, you can see a list of organizational units (OUs) and their allow list status. By default, all OUs are on the allow list, which means that accounts can be provisioned under them. You can disable certain OUs for account provisioning through AWS Service Catalog.

You can view the Amazon VPC configuration options available to your end users when they provision new accounts.

To configure Amazon VPC settings in Account Factory

1. As a central cloud administrator, sign into the AWS Control Tower console with administrator permissions in the master account.
2. From the left side of the dashboard, select Account Factory to navigate to the Account Factory network configuration page. There you can see the default network settings displayed. To edit, select Edit and view the editable version of your Account Factory network configuration settings.
3. You can modify each field of the default settings as needed. Choose the VPC configuration options you’d like to establish for all new Account Factory accounts that your end users may create, and enter your settings into the fields.

• Choose disabled or enabled to create a public subnet in Amazon VPC. By default, the internet-accessible subnet is disallowed.

  Note
  If you set the account factory VPC configuration so that public subnets are enabled when provisioning a new account, account factory configures Amazon VPC to create a NAT Gateway. You will be billed for your usage by Amazon VPC. See VPC Pricing for more information.

• Choose the maximum number of private subnets in Amazon VPC from the list. By default, 1 is selected. The maximum number of private subnets allowed is 2.

• Enter the range of IP addresses for creating your account VPCs. The value must be in the form of a classless inter-domain routing (CIDR) block (for example, the default is 172.31.0.0/16). This CIDR block provides the overall range of subnet IP addresses for the VPC that Account Factory creates for your account. Within your VPC, subnets are assigned automatically from the range you specify, and they are equal in size. By default, subnets within your VPC do not overlap. However, subnet IP address ranges in the VPCs of all your provisioned accounts could overlap.
• Choose a region or all the regions for creating a VPC when an account is provisioned. By default all available regions are selected.
• From the list, choose the number of Availability Zones to configure subnets for in each VPC. The default and recommended number is 3.
• Choose Save.

You can set up these configuration options to create new accounts that don't include a VPC. See the walkthrough.

Unmanaging a Member Account

If you created an account in Account Factory that you no longer want to be managed by AWS Control Tower in a landing zone, you can unmanage the account. This can be done in the AWS Service Catalog console by an AWS SSO user in either the AWSAccountFactory or AWSServiceCatalogAdmins groups. For more information on AWS SSO users or groups, see Managing Users and Access Through AWS Single Sign-On (p. 58). The following procedure describes how to unmanage a member account.

To unmanage a member account

2. From the left navigation pane, choose Provisioned products list.
3. From the list of provisioned accounts, choose the name of the account that you want AWS Control Tower to no longer manage.
4. On the Provisioned product details page, from the Actions menu, choose Terminate.
5. From the dialog box that appears, choose Terminate.

Important
The word terminate is specific to AWS Service Catalog. When you terminate an Account Factory account in AWS Service Catalog, the account is not closed. This action removes the account from its OU and your landing zone.

6. The Deregistering Managed Account message displays.
7. To update the displayed account status, refresh the page. When the account has been unmanaged, its status changes to terminated.
8. If you no longer need the terminated account, close it. For information about closing AWS accounts, see Closing an Account in the AWS Billing and Cost Management User Guide

Note
An unmanaged (terminated) account is not closed or deleted. When the account has been unmanaged, the AWS SSO user that you selected when you created the account in Account Factory still has administrative access to the account. If you do not want this user to have administrative access, you must change this setting in AWS SSO by updating the account in Account Factory and changing the AWS SSO user email address for the account. For more information, see Updating and Moving Account Factory Accounts (p. 66).

Closing an Account Created in Account Factory

Accounts created in Account Factory are AWS accounts. For information about closing AWS accounts, see Closing an Account in the AWS Billing and Cost Management User Guide.
Resource Considerations for Account Factory

When an account is provisioned with Account Factory, the following AWS resources are created within the account.

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS CloudFormation</td>
<td>Stacks</td>
<td>StackSet-AWSControlTowerBP-BASELINE-CLOUDTRAIL-*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-CLOUDWATCH-*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-CONFIG-*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-ROLES-*</td>
</tr>
<tr>
<td></td>
<td></td>
<td>StackSet-AWSControlTowerBP-BASELINE-SERVICE-ROLES-*</td>
</tr>
<tr>
<td>AWS CloudTrail</td>
<td>Trail</td>
<td>aws-controltower-BaselineCloudTrail</td>
</tr>
<tr>
<td>Amazon CloudWatch</td>
<td>CloudWatch Event Rules</td>
<td>aws-controltower-ConfigComplianceChangeEventRule</td>
</tr>
<tr>
<td>Amazon CloudWatch</td>
<td>CloudWatch Logs</td>
<td>aws-controltower/CloudTrailLogs</td>
</tr>
<tr>
<td></td>
<td></td>
<td>/aws/lambda/aws-controltower-NotificationForwarder</td>
</tr>
<tr>
<td>AWS Identity and Access Management</td>
<td>Roles</td>
<td>aws-controltower-AdministratorExecutionRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-CloudWatchLogsRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-ConfigRecorderRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-ForwardSnsNotificationRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>aws-controltower-ReadOnlyExecutionRole</td>
</tr>
<tr>
<td></td>
<td></td>
<td>AWSControlTowerExecution</td>
</tr>
<tr>
<td>AWS Identity and Access Management</td>
<td>Policies</td>
<td>AWSControlTowerServiceRolePolicy</td>
</tr>
<tr>
<td>Amazon Simple Notification Service</td>
<td>Topics</td>
<td>aws-controltower-SecurityNotifications</td>
</tr>
<tr>
<td>AWS Lambda</td>
<td>Applications</td>
<td>StackSet-AWSControlTowerBP-BASELINE-CLOUDWATCH-*</td>
</tr>
</tbody>
</table>
Guardrails and Account Resources

Accounts created through the Account Factory in AWS Control Tower inherit the guardrails of the parent OU, and the associated resources are created.

**Note**
Accounts created outside of AWS Control Tower won't inherit guardrails from the parent OU in AWS Control Tower. These accounts also do not display in AWS Control Tower.

When you enable guardrails with strongly recommended guidance, AWS Control Tower creates and manages certain additional AWS resources in your accounts. Do not modify or delete resources created by AWS Control Tower. Doing so could result in the guardrails entering an unknown state. For more information, see Guardrail Reference (p. 32).

<table>
<thead>
<tr>
<th>AWS service</th>
<th>Resource type</th>
<th>Resource name</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Lambda</td>
<td>Functions</td>
<td>aws-controltower-NotificationForwarder</td>
</tr>
</tbody>
</table>
Detecting and Resolving Drift in AWS Control Tower

Identifying and resolving drift is a regular operations task for AWS Control Tower master account administrators.

When you create your landing zone, the landing zone and all the OUs, accounts, and resources are compliant with the governance rules enforced by your chosen guardrails. As you and your organization members use the landing zone, changes in this compliance status may occur. Some changes may be accidental, and some may be made intentionally to respond to time-sensitive operational events.

Changes can complicate your compliance story. Drift detection assists you in identifying resources that need changes or configuration updates to resolve the drift. Resolving drift helps to ensure your compliance with governance regulations.

Detecting Drift

Drift is detected automatically by AWS Control Tower. It is surfaced in the Amazon SNS notifications that are aggregated in the audit account. Notifications in each member account send alerts to a local Amazon SNS topic, and to a Lambda function.

Member account administrators can (and as a best practice, they should) subscribe to the SNS drift notifications for specific accounts. The AWS Control Tower console indicates to master account administrators when drift has occurred.

Resolving Drift

Although detection is automatic, the steps to resolve drift must be done through the console. Many types of drift can be resolved through the Settings page. If the Repair button in the Versions section of the page is selectable, you can choose Repair to repair some types of drift. If no drift has occurred, the Repair button appears greyed-out.

Most types of drift can be resolved by administrators. A few types of drift must be repaired immediately, including deletion of an organizational unit that the AWS Control Tower landing zone requires. Here are some examples of major drift:

- The organizational unit originally named Core during landing zone setup by AWS Control Tower should not be deleted. If you delete it, you'll see an error message instructing you to repair the landing zone immediately. You won't be able to take any other actions in AWS Control Tower until the repair is complete.
- AWS Control Tower checks certain IAM roles when you log into the console for IAM role drift. If these roles are missing or inaccessible, you'll see an error page instructing you to repair your landing zone. These roles are AWSControlTowerAdmin, AWSControlTowerCloudTrailRole, and AWSControlTowerStackSetRole.
- If you delete the organizational unit originally named Custom during landing zone setup by AWS Control Tower, your landing zone will be in a state of drift, but you still can use AWS Control Tower. At least one non-Core OU is required for AWS Control Tower to operate, but it doesn't have to be the Custom OU.

Repairable Changes to Resources

Here's a list of changes to AWS Control Tower resources that are permitted, although they create repairable drift. Results of these permitted operations are viewable in the AWS Control Tower console, although a refresh may be required.
For more information about how to resolve the resulting drift, see Managing Resources Outside of AWS Control Tower.

**Changes Permitted Outside the AWS Control Tower Console**

- Change the name of a non-Core OU.
- Change the name of the Core OU.
- Change the name of non-Core (member) accounts.
- Change the name of Core (shared) accounts.
- Delete a non-Core OU.
- Delete a non-Core OU (member) account.
- Change the email address of a Core (shared) account.
- Change the email address of a non-core (member) account.

**Note**
Moving accounts between OUs is considered drift, and it must be repaired.

**Drift and New Account Provisioning**

If your landing zone is in a state of drift, the Enroll account feature in AWS Control Tower will not work. In that case, you must provision new accounts through AWS Service Catalog. For instructions, see Provisioning Account Factory Accounts With AWS Service Catalog (p. 65).

In particular, if you've made certain changes to your accounts by means of AWS Service Catalog, such as changing the name of your portfolio, the Enroll account feature will not work.

**Types of Governance Drift**

Governance drift occurs when organizations and member accounts are changed and updated. The types of governance drift that can be detected in AWS Control Tower are as follows:

- Moved Member Account (p. 72)
- Added Member Account (p. 73)
- Removed Member Account (p. 73)
- Unplanned Update to Managed SCP (p. 74)
- SCP Attached to Member Account (p. 75)
- SCP Detached from Managed OU (p. 75)
- Deleted Managed OU (p. 76)

**Moved Member Account**

This kind of drift can occur when a member account, the audit account, or the log archive account is moved from one AWS Control Tower OU to another AWS Control Tower OU. The following is an example of the Amazon SNS notification when this type of drift is detected.

```json
{
  "Message" : "AWS Control Tower has detected that your managed account 'account@email@amazon.com (012345678909)' has been moved from organizational unit 'Custom (ou-0123-
```
Resolutions

When this kind of drift occurs, you can resolve it as follows:

- **Account Factory Provisioned Account** – You can resolve the drift by updating the account in Account Factory. For more information, see Updating and Moving Account Factory Accounts (p. 66).
- **Shared account** – You can resolve the drift from moving the audit or log archive account by updating your landing zone. For more information, see Update Your Landing Zone (p. 100).

### Added Member Account

This kind of drift can occur when an AWS Control Tower member account is added to an AWS Control Tower OU. The following is an example of the Amazon SNS notification when this type of drift is detected.

```
{
  "Message": "AWS Control Tower has detected that the managed account 'account-email@amazon.com (012345678909)' has been added to organization o-123EXAMPLE. For more information, including steps to resolve this issue, see 'https://docs.aws.amazon.com/console/controltower/add-account'",
  "MasterAccountId": "012345678912",
  "OrganizationId": "o-123EXAMPLE",
  "DriftType": "AccountAddedToOrganization",
  "RemediationStep": "Update Account Factory Provisioned Product",
  "AccountId": "012345678909",
  "SourceId": "012345678909",
  "DestinationId": "ou-3210-1EXAMPLE"
}
```

### Resolution

When this kind of drift occurs, you can resolve it by updating the account in Account Factory. For more information, see Updating and Moving Account Factory Accounts (p. 66).

For more information about resolving drift for accounts and OUs, see Managing Resources Outside of AWS Control Tower (p. 26).

### Removed Member Account

This kind of drift can occur when an AWS Control Tower member account is removed from an AWS Control Tower OU. The following is an example of the Amazon SNS notification when this type of drift is detected.

```
{
  "Message": "AWS Control Tower has detected that the managed account 012345678909 has been removed from organization o-123EXAMPLE. For more information, including steps to resolve this issue, see 'https://docs.aws.amazon.com/console/controltower/remove-account'",
  "MasterAccountId": "012345678912",
  "OrganizationId": "o-123EXAMPLE",
  "DriftType": "AccountRemovedFromOrganization",
  "RemediationStep": "Update Account Factory Provisioned Product",
  "AccountId": "012345678909",
  "SourceId": "012345678909",
  "DestinationId": "ou-3210-1EXAMPLE"
}
```
Resolution

When this kind of drift occurs, AWS Control Tower receives a notification and makes an automatic update. If the deleted member account continues to be displayed, you can resolve the drift by repairing your landing zone. You also could resolve the drift by updating the account in Account Factory, and adding the account to another registered OU from the Account Factory update wizard. For more information, see Updating and Moving Account Factory Accounts (p. 66).

For more information about resolving drift for accounts and OUs, see Managing Resources Outside of AWS Control Tower (p. 26).

Note

In AWS Service Catalog, the Account Factory provisioned product that represents the account is not updated to remove the account. Instead, the provisioned product is displayed as Tainted and in an error state. To clean up, go to the AWS Service Catalog, choose the provisioned product, and then choose Terminate.

Unplanned Update to Managed SCP

This kind of drift can occur when an SCP for a guardrail is updated in the Organizations console or programmatically using the AWS CLI or one of the AWS SDKs. The following is an example of the Amazon SNS notification when this type of drift is detected.

"{
  "Message" : "AWS Control Tower has detected that the managed service control policy 'aws-guardrails-012345 (p-tEXAMPLE)', attached to the managed organizational unit 'Core (ou-0123-1EXAMPLE)', has been modified. For more information, including steps to resolve this issue, see 'https://docs.aws.amazon.com/console/controllower/update-scp''",
  "MasterAccountId" : "012345678912",
  "OrganizationId" : "o-123EXAMPLE",
  "DriftType" : "ServiceControlPolicyUpdated",
  "RemediationStep" : "Update Control Tower Setup",
  "OrganizationalUnitId" : "ou-0123-1EXAMPLE",
  "PolicyId" : "p-tEXAMPLE"
}

Resolution

When this kind of drift occurs, you can resolve it by updating your landing zone. For more information, see Update Your Landing Zone (p. 100).

SCP Attached to Managed OU

This kind of drift can occur when an SCP is attached to an OU outside of the AWS Control Tower console. The following is an example of the Amazon SNS notification when this type of drift is detected.
**Resolution**

When this kind of drift occurs, you can resolve it by updating your landing zone. For more information, see Update Your Landing Zone (p. 100).

### SCP Detached from Managed OU

This kind of drift can occur when an SCP has been detached from an OU outside of the AWS Control Tower console. The following is an example of the Amazon SNS notification when this type of drift is detected.

```
{""Message"": "AWS Control Tower has detected that the managed service control policy 'aws-guardrails-012345 (p-tEXAMPLE)' has been detached from the managed organizational unit 'Custom (ou-0123-1EXAMPLE)'. For more information, including steps to resolve this issue, see 'https://docs.aws.amazon.com/console/controltower/scp-detached'",
""MasterAccountId"": "012345678912",
""OrganizationId"": "ou-123EXAMPLE",
""DriftType"": "ServiceControlPolicyDetachedFromOrganizationalUnit",
""RemediationStep"": "Update Control Tower Setup",
""OrganizationalUnitId"": "ou-0123-1EXAMPLE",
""PolicyId"": "p-tEXAMPLE"}
```

**Resolution**

When this kind of drift occurs, you can resolve it by updating your landing zone. For more information, see Update Your Landing Zone (p. 100).

### SCP Attached to Member Account

This kind of drift can occur when an SCP is attached to an account in the Organizations console. Guardrails and their SCPs can be enabled on OUs and all of an OU's member accounts through the AWS Control Tower console. The following is an example of the Amazon SNS notification when this type of drift is detected.

```
{""Message"": "AWS Control Tower has detected that the managed service control policy 'aws-guardrails-012345 (p-tEXAMPLE)' has been attached to the managed account 'account-email@amazon.com (012345678909)'. For more information, including steps to resolve this issue, see 'https://docs.aws.amazon.com/console/controltower/scp-detached-account'",
""MasterAccountId"": "012345678912",
""OrganizationId"": "ou-123EXAMPLE",
""DriftType"": "ServiceControlPolicyAttachedToOrganizationalUnit",
""RemediationStep"": "Update Control Tower Setup",
""OrganizationalUnitId"": "ou-0123-1EXAMPLE",
""PolicyId"": "p-tEXAMPLE"}
```
Resolution

When this kind of drift occurs, you can resolve it by updating your landing zone. For more information, see Update Your Landing Zone (p. 100).

Deleted Managed OU

This kind of drift can occur if an AWS Control Tower OU is deleted outside of the AWS Control Tower console. The following is an example of the Amazon SNS notification when this type of drift is detected.

```
{""Message"" : "AWS Control Tower has detected that the managed organizational unit 'Custom (ou-0123-1EXAMPLE) has been deleted. For more information, including steps to resolve this issue, see 'https://docs.aws.amazon.com/console/controltower/delete-ou'"",
 ""MasterAccountId"" : "012345678912",
 ""OrganizationId"" : "o-123EXAMPLE",
 ""DriftType"" : "OrganizationalUnitDeleted",
 ""RemediationStep"" : "Delete managed organizational unit in Control Tower",
 ""OrganizationalUnitId"" : "ou-0123-1EXAMPLE"
}
```

Resolution

When this kind of drift occurs, AWS Control Tower usually can update the list of registered OUs automatically. If a deleted OU is shown, a central cloud administrator must sign in to the AWS Control Tower console and delete the OU from your list of Organizational units.

For more information about resolving drift for accounts and OUs, see Managing Resources Outside of AWS Control Tower (p. 26).
Security in AWS Control Tower

Cloud security at AWS is the highest priority. As an AWS customer, you benefit from a data center and network architecture that is built to meet the requirements of the most security-sensitive organizations.

Security is a shared responsibility between AWS and you. The shared responsibility model describes this as security of the cloud and security in the cloud:

- **Security of the cloud** – AWS is responsible for protecting the infrastructure that runs AWS services in the AWS Cloud. AWS also provides you with services that you can use securely. The effectiveness of our security is regularly tested and verified by third-party auditors as part of the AWS compliance programs. To learn about the compliance programs that apply to AWS Control Tower, see AWS Services in Scope by Compliance Program.

- **Security in the cloud** – Your responsibility is determined by the AWS services that you use. You are also responsible for other factors including the sensitivity of your data, your organization's requirements, and applicable laws and regulations.

This documentation helps you understand how to apply the shared responsibility model when using AWS Control Tower. The following topics show you how to configure AWS Control Tower to meet your security and compliance objectives. You also learn how to use other AWS services that help you monitor and secure your AWS Control Tower resources.

Data Protection in AWS Control Tower

AWS Control Tower conforms to the AWS shared responsibility model, which includes regulations and guidelines for data protection. AWS is responsible for protecting the global infrastructure that runs all the AWS services. AWS maintains control over data hosted on this infrastructure, including the security configuration controls for handling customer content and personal data. AWS customers and APN partners, acting either as data controllers or data processors, are responsible for any personal data that they put in the AWS Cloud.

For data protection purposes, we recommend that you protect AWS account credentials and set up individual user accounts with AWS Identity and Access Management (IAM), so that each user is given only the permissions necessary to fulfill their job duties. We also recommend that you secure your data in the following ways:

- Use multi-factor authentication (MFA) with each account.
- Use SSL/TLS to communicate with AWS resources.
- Set up API and user activity logging with CloudTrail. This is handled automatically in AWS Control Tower when you set up your landing zone.
- Use AWS encryption solutions, along with all default security controls within AWS services.
- Use advanced managed security services such as Amazon Macie, which assists in discovering and securing personal data that is stored in Amazon S3.

We strongly recommend that you never put sensitive identifying information, such as your customers' account numbers, into free-form fields such as a Name field. This includes when you work with AWS Control Tower or other AWS services using the console, API, AWS CLI, or AWS SDKs. Any data that you enter into AWS Control Tower or other services might get picked up for inclusion in diagnostic logs. When you provide a URL to an external server, don't include credentials information in the URL to validate your request to that server.
For more information about data protection, see the AWS Shared Responsibility Model and GDPR blog post on the AWS Security Blog. AWS Control Tower provides the following options that you can use to help secure the content that exists in your landing zone:

**Topics**
- Encryption at Rest (p. 78)
- Encryption in Transit (p. 78)
- Restrict Access to Content (p. 78)

**Encryption at Rest**

AWS Control Tower uses Amazon S3 buckets and Amazon DynamoDB databases that are encrypted at rest by using Amazon S3-Managed Keys (SSE-S3) in support of your landing zone. This encryption is configured by default when you set up your landing zone. You can also establish encryption at rest for the services you use in your landing zone for the services that support it. For more information, see the security chapter of that service's online documentation.

**Encryption in Transit**

AWS Control Tower uses Transport Layer Security (TLS) and client-side encryption for encryption in transit in support of your landing zone. In addition, accessing AWS Control Tower requires using the console, which can only be accessed through an HTTPS endpoint. This encryption is configured by default when you set up your landing zone.

**Restrict Access to Content**

As a best practice, you should restrict access to the appropriate subset of users. With AWS Control Tower, you can do this by ensuring your central cloud administrators and end users have the right IAM permissions or, in the case of AWS SSO users, are in the correct groups.

- For more information about roles and policies for IAM entities, see IAM User Guide.
- For more information about the AWS SSO groups that are created when you set up your landing zone, see AWS SSO Groups for AWS Control Tower (p. 60).

**Identity and Access Management in AWS Control Tower**

To perform any operation in your landing zone, such as provisioning accounts in Account Factory or creating new organizational units (OUs) in the AWS Control Tower console, either AWS Identity and Access Management (IAM) or AWS Single Sign-On (AWS SSO) require that you authenticate that you're an approved AWS user. For example, if you're using the AWS Control Tower console, you authenticate your identity by providing your AWS user name and a password.

After you authenticate your identity, IAM controls your access to AWS with a defined set of permissions on a specific set of operations and resources. If you are an account administrator, you can use IAM to control the access of other IAM users to the resources that are associated with your account.

**Topics**
- Authentication (p. 79)
- Access Control (p. 80)
- Overview of Managing Access Permissions to Your AWS Control Tower Resources (p. 80)
Authentication

You have access to AWS as any of the following types of identities:

- **AWS account root user** – When you first create an AWS account, you begin with an identity that has complete access to all AWS services and resources in the account. This identity is called the AWS account root user. You have access to this identity when you sign in with the email address and password that you used to create the account. We strongly recommend that you do not use the root user for your everyday tasks, even the administrative ones. Instead, adhere to the best practice of using the root user only to create your first IAM user. Then securely lock away the root user credentials and use them to perform only a few account and service management tasks. For more information, see Sign in as a Root User (p. 23).

- **IAM user** – An IAM user is an identity within your AWS account that has specific custom permissions. You can use an IAM user name and password to sign in to secure AWS webpages like the AWS Management Console, AWS Discussion Forums, or the AWS Support Center.

  In addition to a user name and password, you can also generate access keys for each user. You can use these keys when you access AWS services programmatically, either through one of the several SDKs or by using the AWS Command Line Interface (CLI). The SDK and CLI tools use the access keys to cryptographically sign your request. If you don’t use AWS tools, you must sign the request yourself. AWS Control Tower supports Signature Version 4, a protocol for authenticating inbound API requests. For more information about authenticating requests, see Signature Version 4 Signing Process in the AWS General Reference.

- **IAM role** – An IAM role is an IAM identity that you can create in your account that has specific permissions. An IAM role is similar to an IAM user in that it is an AWS identity with permissions policies that determine what the identity can and cannot do in AWS. However, instead of being uniquely associated with one person, a role is intended to be assumable by anyone who needs it. Also, a role does not have standard long-term credentials such as a password or access keys associated with it. Instead, when you assume a role, it provides you with temporary security credentials for your role session. IAM roles with temporary credentials are useful in the following situations:

  - **Federated user access** – Instead of creating an IAM user, you can use existing identities from AWS Directory Service, your enterprise user directory, or a web identity provider. These are known as federated users. AWS assigns a role to a federated user when access is requested through an identity provider. For more information about federated users, see Federated Users and Roles in the IAM User Guide.

  - **AWS service access** – A service role is an IAM role that a service assumes to perform actions in your account on your behalf. When you set up some AWS service environments, you must define a role for the service to assume. This service role must include all the permissions that are required for the service to access the AWS resources that it needs. Service roles vary from service to service, but many allow you to choose your permissions as long as you meet the documented requirements for that service. Service roles provide access only within your account and cannot be used to grant access to services in other accounts. You can create, modify, and delete a service role from within IAM. For example, you can create a role that allows Amazon Redshift to access an Amazon S3 bucket on your behalf and then load data from that bucket into an Amazon Redshift cluster. For more information, see Creating a Role to Delegate Permissions to an AWS Service in the IAM User Guide.

  - **Applications running on Amazon EC2** – You can use an IAM role to manage temporary credentials for applications that are running on an Amazon EC2 instance and making AWS CLI or AWS API requests. This is preferable to storing access keys within the Amazon EC2 instance. To assign an AWS role to an Amazon EC2 instance and make it available to all of its applications, you create an instance profile that is attached to the instance. An instance profile contains the role and enables programs that are running on the Amazon EC2 instance to get temporary credentials. For more information, see Using an IAM Role to Grant Permissions to Applications Running on Amazon EC2 Instances in the IAM User Guide.
• **AWS SSO user** Authentication to the AWS SSO user portal is controlled by the directory that you have connected to AWS SSO. However, authorization to the AWS accounts that are available to end users from within the user portal is determined by two factors:
  
  • Who has been assigned access to those AWS accounts in the AWS SSO console. For more information, see *Single Sign-On Access* in the *AWS Single Sign-On User Guide*.
  
  • What level of permissions have been granted to the end users in the AWS SSO console to allow them the appropriate access to those AWS accounts. For more information, see *Permission Sets* in the *AWS Single Sign-On User Guide*.

## Access Control

To create, update, delete, or list AWS Control Tower resources, or other AWS resources in your landing zone you need permissions to perform the operation, and you need permissions to access the corresponding resources. In addition, to perform the operation programmatically, you need valid access keys.

The following sections describe how to manage permissions for AWS Control Tower:

### Topics

- Overview of Managing Access Permissions to Your AWS Control Tower Resources (p. 80)
- Using Identity-Based Policies (IAM Policies) for AWS Control Tower (p. 83)

## Overview of Managing Access Permissions to Your AWS Control Tower Resources

Every AWS resource is owned by an AWS account, and permissions to create or access a resource are governed by permissions policies. An account administrator can attach permissions policies to IAM identities (that is, users, groups, and roles), and some services (such as AWS Lambda) also support attaching permissions policies to resources.

**Note**

An *account administrator* (or administrator) is a user with administrator privileges. For more information, see *IAM Best Practices* in the *IAM User Guide*.

When granting permissions, you decide who is getting the permissions, the resources they get permissions for, and the specific actions that you want to allow on those resources.

### Topics

- AWS Control Tower Resources and Operations (p. 80)
- Understanding Resource Ownership (p. 81)
- Managing Access to Resources (p. 81)
- Specifying Policy Elements: Actions, Effects, and Principals (p. 82)
- Specifying Conditions in a Policy (p. 82)

## AWS Control Tower Resources and Operations

In AWS Control Tower, the primary resource is a *landing zone*. AWS Control Tower also supports an additional resource type, *guardrails*. However, for AWS Control Tower, you can manage guardrails only in the context of an existing landing zone. Guardrails are referred to as a *subresource*. 
Understanding Resource Ownership

The AWS account owns the resources that are created in the account, regardless of who created the resources. Specifically, the resource owner is the AWS account of the principal entity (that is, the AWS account root user, an IAM user, or an IAM role) that authenticates the resource creation request. The following examples illustrate how this works:

- If you use the AWS account root user credentials of your AWS account to set up a landing zone, your AWS account is the owner of the resource.
- If you create an IAM user in your AWS account and grant permissions to set up a landing zone to that user, the user can set up a landing zone as long as their account meets the prerequisites. However, your AWS account, to which the user belongs, owns the landing zone resource.
- If you create an IAM role in your AWS account with permissions to set up a landing zone, anyone who can assume the role can set up a landing zone. Your AWS account, to which the role belongs, owns the landing zone resource.

Managing Access to Resources

A permissions policy describes who has access to what. The following section explains the available options for creating permissions policies.

Note
This section discusses using IAM in the context of AWS Control Tower. It doesn't provide detailed information about the IAM service. For complete IAM documentation, see What Is IAM? in the IAM User Guide. For information about IAM policy syntax and descriptions, see AWS IAM Policy Reference in the IAM User Guide.

Policies attached to an IAM identity are referred to as identity-based policies (IAM policies). Policies attached to a resource are referred to as resource-based policies. AWS Control Tower supports only identity-based policies (IAM policies).

Topics
- Identity-Based Policies (IAM Policies) (p. 81)
- Resource-Based Policies (p. 82)

Identity-Based Policies (IAM Policies)

You can attach policies to IAM identities. For example, you can do the following:

- **Attach a permissions policy to a user or a group in your account** – To grant a user permissions to create an AWS Control Tower resource, such as setting up a landing zone, you can attach a permissions policy to a user or group that the user belongs to.

- **Attach a permissions policy to a role (grant cross-account permissions)** – You can attach an identity-based permissions policy to an IAM role to grant cross-account permissions. For example, the administrator in Account A can create a role to grant cross-account permissions to another AWS account (for example, Account B) or an AWS service as follows:
  1. Account A administrator creates an IAM role and attaches a permissions policy to the role that grants permissions on resources in Account A.
  2. Account A administrator attaches a trust policy to the role identifying Account B as the principal who can assume the role.
  3. Account B administrator can then delegate permissions to assume the role to any users in Account B. Doing this allows users in Account B to create or access resources in Account A. The principal in the trust policy can also be an AWS service principal if you want to grant an AWS service permissions to assume the role.
For more information about using IAM to delegate permissions, see Access Management in the IAM User Guide.

The following is an example policy that allows a user to set up a landing zone in your AWS account.

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

For more information about using identity-based policies with AWS Control Tower, see Using Identity-Based Policies (IAM Policies) for AWS Control Tower (p. 83). For more information about users, groups, roles, and permissions, see Identities (Users, Groups, and Roles) in the IAM User Guide.

Resource-Based Policies

Other services, such as Amazon S3, also support resource-based permissions policies. For example, you can attach a policy to an S3 bucket to manage access permissions to that bucket.

Specifying Policy Elements: Actions, Effects, and Principals

Currently, AWS Control Tower doesn't have an API. You can set up and manage your landing zone through the AWS Control Tower console. To set up your landing zone, you must be an IAM user with administrative permissions as defined in a IAM policy.

The following are the most basic policy elements:

- **Resource** – In a policy, you use an Amazon Resource Name (ARN) to identify the resource to which the policy applies. For more information, see AWS Control Tower Resources and Operations (p. 80).
- **Action** – You use action keywords to identify resource operations that you want to allow or deny.
- **Effect** – You specify the effect when the user requests the specific action—this can be either allow or deny. If you don't explicitly grant access to (allow) a resource, access is implicitly denied. You can also explicitly deny access to a resource, which you might do to make sure that a user cannot access it, even if a different policy grants access.
- **Principal** – In identity-based policies (IAM policies), the user that the policy is attached to is the implicit principal. For resource-based policies, you specify the user, account, service, or other entity that you want to receive permissions (applies to resource-based policies only). AWS Control Tower doesn't support resource-based policies.

To learn more about IAM policy syntax and descriptions, see AWS IAM Policy Reference in the IAM User Guide.

Specifying Conditions in a Policy

When you grant permissions, you can use the IAM policy language to specify the conditions when a policy should take effect. For example, you might want a policy to be applied only after a specific date. For more information about specifying conditions in a policy language, see Condition in the IAM User Guide.
To express conditions, you use predefined condition keys. There are no condition keys specific to AWS Control Tower. However, there are AWS-wide condition keys that you can use as appropriate. For a complete list of AWS-wide keys, see Available Keys for Conditions in the IAM User Guide.

### Using Identity-Based Policies (IAM Policies) for AWS Control Tower

This topic provides examples of identity-based policies that demonstrate how an account administrator can attach permissions policies to IAM identities (that is, users, groups, and roles) and thereby grant permissions to perform operations on AWS Control Tower resources.

**Important**

We recommend that you first review the introductory topics that explain the basic concepts and options available for you to manage access to your AWS Control Tower resources. For more information, see Overview of Managing Access Permissions to Your AWS Control Tower Resources (p. 80).

The following shows an example of a permissions policy.

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

The policy has one statement that grants permissions for all AWS actions on all resources in the account. This is the permissions policy for administrator access in an AWS account. This is the necessary level of permissions for an IAM entity that will set up a landing zone.

The policy doesn't specify the Principal element because in an identity-based policy you don't specify the principal who gets the permission. When you attach policy to a user, the user is the implicit principal. When you attach a permissions policy to an IAM role, the principal identified in the role's trust policy gets the permissions.

### Permissions Required to Use the AWS Control Tower Console

AWS Control Tower requires creation of three roles to set up a landing zone. AWS Control Tower splits permissions into three roles as a best practice to restrict access to the minimal sets of actions and resources.

**AWSControlTowerAdmin**

This role provides AWS Control Tower with access to infrastructure critical to maintaining the landing zone. The role requires an inline policy, and a managed policy attachment.

**Managed Policy: AWSControlTowerServiceRolePolicy**

**Inline Policy:**

```json
{
    "Version": "2012-10-17",
    "Statement": [
```
AWS Control Tower User Guide
Logging and Monitoring

Logging and Monitoring in AWS Control Tower

Monitoring allows you to plan for and respond to potential incidents. Therefore, monitoring is an important part of the well-architected nature of AWS Control Tower. The results of monitoring activities are stored in log files; therefore, logging and monitoring are closely related concepts.

When you set up your landing zone, one of the shared accounts created is the log archive account, dedicated to collecting all logs centrally, including logs for all of your other accounts. These log files allow administrators and auditors to review actions and events that have occurred.
As a best practice, you should collect monitoring data from all of the parts of your AWS solution into your logs, so that you can more easily debug a multi-point failure if one occurs. AWS provides several tools for monitoring your resources and activity in your landing zone.

For example, the status of your guardrails is monitored constantly. You can see their status at a glance in the AWS Control Tower console. The health and status of the accounts you provisioned in Account Factory also is monitored constantly.

Logging

Logging of actions and events in AWS Control Tower is accomplished automatically through its integration with CloudWatch. All actions are logged, including actions from the AWS Control Tower master accounts and from your organization's member accounts. Master account actions and events are viewable on the Activities page in the console. Member account actions and events are viewable in log archive files.

The Activities Page

The Activities page provides an overview of AWS Control Tower master account actions. To navigate to the AWS Control Tower Activities page, select Activities from the left navigation.

The Activities page shows all AWS Control Tower actions initiated from the master account. It includes actions that are logged automatically when you navigate through the AWS Control Tower console. Here are the fields that the Activities page shows you:

- Date and time: The timestamp for the activity.
- User: The person or account that initiated the activity.
- Action: The activity that occurred.
- Resources: The resources affected by the activity.
- Status: Success, failure, or other state of the activity.
- Description: More details about the activity.

The activities shown in the Activities page are the same ones reported in the AWS CloudTrail events log for AWS Control Tower, but they're shown in a table format. To learn more about a specific activity, select the activity from the table and then choose View details.

The following sections describe monitoring and logging in AWS Control Tower with more detail:

Topics

- Monitoring (p. 85)
- Logging AWS Control Tower Actions with AWS CloudTrail (p. 86)

Monitoring

Monitoring is an important part of maintaining the reliability, availability, and performance of AWS Control Tower and your other AWS solutions. AWS provides the following monitoring tools to watch AWS Control Tower, report when something is wrong, and take automatic actions when appropriate:

- Amazon CloudWatch monitors your AWS resources and the applications you run on AWS in real time. You can collect and track metrics, create customized dashboards, and set alarms that notify you or take actions when a specified metric reaches a threshold that you specify. For example, you can have CloudWatch track CPU usage or other metrics of your Amazon EC2 instances and automatically launch new instances when needed. For more information, see the Amazon CloudWatch User Guide.
• *Amazon CloudWatch Events* delivers a near real-time stream of system events that describe changes in AWS resources. CloudWatch Events enables automated event-driven computing, as you can write rules that watch for certain events and trigger automated actions in other AWS services when these events happen. For more information, see the *Amazon CloudWatch Events User Guide*.

• *Amazon CloudWatch Logs* enables you to monitor, store, and access your log files from Amazon EC2 instances, CloudTrail, and other sources. CloudWatch Logs can monitor information in the log files and notify you when certain thresholds are met. You can also archive your log data in highly durable storage. For more information, see the *Amazon CloudWatch Logs User Guide*.

• *AWS CloudTrail* captures API calls and related events made by or on behalf of your AWS account and delivers the log files to an Amazon S3 bucket that you specify. You can identify which users and accounts called AWS, the source IP address from which the calls were made, and when the calls occurred.

For more information, see Logging AWS Control Tower Actions with AWS CloudTrail (p. 86).

### Logging AWS Control Tower Actions with AWS CloudTrail

AWS Control Tower is integrated with AWS CloudTrail, a service that provides a record of actions taken by a user, role, or an AWS service in AWS Control Tower. CloudTrail captures actions for AWS Control Tower as events. If you create a trail, you can enable continuous delivery of CloudTrail events to an Amazon S3 bucket, including events for AWS Control Tower. If you don't configure a trail, you can still view the most recent events in the CloudTrail console in *Event history*. Using the information collected by CloudTrail, you can determine the request that was made to AWS Control Tower, the IP address from which the request was made, who made the request, when it was made, and additional details.

To learn more about CloudTrail, including how to configure and enable it, see the *AWS CloudTrail User Guide*.

### AWS Control Tower Information in CloudTrail

CloudTrail is enabled on your AWS account when you create the account. When supported event activity occurs in AWS Control Tower, that activity is recorded in a CloudTrail event along with other AWS service events in *Event history*. You can view, search, and download recent events in your AWS account. For more information, see *Viewing Events with CloudTrail Event History*.

For an ongoing record of events in your AWS account, including events for AWS Control Tower, create a trail. A trail enables CloudTrail to deliver log files to an Amazon S3 bucket. By default, when you create a trail in the console, the trail applies to all AWS Regions. The trail logs events from all Regions in the AWS partition and delivers the log files to the Amazon S3 bucket that you specify. Additionally, you can configure other AWS services to further analyze and act upon the event data collected in CloudTrail logs. For more information, see the following:

- Overview for Creating a Trail
- CloudTrail Supported Services and Integrations
- Configuring Amazon SNS Notifications for CloudTrail
- Receiving CloudTrail Log Files from Multiple Regions and Receiving CloudTrail Log Files from Multiple Accounts

AWS Control Tower logs the following actions as events in CloudTrail log files:

- SetupLandingZone
• UpdateAccountFactoryConfig
• ManageOrganizationalUnit
• CreateManagedAccount
• EnableGuardrail
• GetLandingZoneStatus
• GetHomeRegion
• ListManagedAccounts
• DescribeManagedAccount
• DescribeAccountFactoryConfig
• DescribeGuardrailForTarget
• DescribeManagedOrganizationalUnit
• ListEnabledGuardrails
• ListGuardrailViolations
• ListGuardrails
• ListGuardrailsForTarget
• ListManagedAccountsForGuardrail
• ListManagedAccountsForParent
• ListManagedOrganizationalUnits
• ListManagedOrganizationalUnitsForGuardrail
• GetGuardrailComplianceStatus
• DescribeGuardrail
• ListDirectoryGroups
• DescribeSingleSignOn
• DescribeCoreService
• GetAvailableUpdates

Every event or log entry contains information about who generated the request. The identity information helps you determine the following:

• Whether the request was made with root or AWS Identity and Access Management (IAM) user credentials.
• Whether the request was made with temporary security credentials for a role or federated user.
• Whether the request was made by another AWS service.

For more information, see the CloudTrail userIdentity Element.

**Example: AWS Control Tower Log File Entries**

A trail is a configuration that enables delivery of events as log files to an Amazon S3 bucket that you specify. CloudTrail log files contain one or more log entries. An event represents a single request from any source and includes information about the requested action, the date and time of the action, request parameters, and so on. CloudTrail events don't appear in any specific order in the log files.

The following example shows a CloudTrail log entry that shows the structure of a typical log file entry for a SetupLandingZone AWS Control Tower event, including a record of the identity of the user who initiated the action.
Lifecycle Events in AWS Control Tower

Some events logged by AWS Control Tower are lifecycle events. A lifecycle event's purpose is to mark the completion of certain AWS Control Tower actions that change the state of resources. Lifecycle events apply to resources that AWS Control Tower creates or manages, such as organizational units (OUs), accounts, and guardrails.

Characteristics of AWS Control Tower lifecycle events

- For each lifecycle event, the event log shows whether the originating Control Tower action completed successfully, or failed.
- AWS CloudTrail automatically records each lifecycle event as a non-API AWS service event. For more information, see the AWS CloudTrail User Guide.
- Each lifecycle event also is delivered to the Amazon EventBridge and Amazon CloudWatch Events services.
Lifecycle events in AWS Control Tower offer two primary benefits:

- Because a lifecycle event registers the completion of an AWS Control Tower action, you can create an Amazon EventBridge rule or Amazon CloudWatch Events rule that can trigger the next steps in your automation workflow, based on the state of the lifecycle event.
- The logs provide additional detail to assist administrators and auditors in reviewing certain types of activity in your organizations.

How lifecycle events work

AWS Control Tower relies upon multiple services to implement its actions. Therefore, each lifecycle event is recorded only after a series of actions is complete. For example, when you enable a guardrail on an OU, AWS Control Tower launches a series of sub-steps that implement the request. The final result of the entire series of sub-steps is recorded in the log as the state of the lifecycle event.

- If every underlying sub-step has completed successfully, the lifecycle event state is recorded as Succeeded.
- If any of the underlying sub-steps did not complete successfully, the lifecycle event state is recorded as Failed.

Each lifecycle event includes a logged timestamp that shows when the AWS Control Tower action was initiated, and another timestamp showing when the lifecycle event is completed, marking success or failure.

Viewing lifecycle events in Control Tower

You can view lifecycle events from the Activities page in your AWS Control Tower dashboard.

- To navigate to the Activities page, choose Activities from the left navigation pane.
- To get more details about a specific event, select the event and then choose the View details button at the upper right.

For more information about how to integrate AWS Control Tower lifecycle events into your workflows, see this blog post, Using lifecycle events to track AWS Control Tower actions and trigger automated workflows.

Lifecycle event names

Each lifecycle event is named so that it corresponds to the originating AWS Control Tower action, which also is recorded by AWS CloudTrail. Thus, for example, a lifecycle event originated by the AWS Control Tower CreateManagedAccount CloudTrail event is named CreateManagedAccount.

Each name in the list that follows is a link to an example of the logged detail in JSON format. The additional detail shown in these examples is taken from the Amazon CloudWatch event logs.

Although JSON does not support comments, some comments have been added in the examples for explanatory purposes. Comments are preceded by "//" and they appear in the right side of the examples.

In these examples, some account names and organization names are obscured. An account ID is always a 12-number sequence, which has been replaced with "xxxxxxxxxxxx" in the examples. An organizational unit ID is a unique string of letters and numbers. Its form is preserved in the examples.

- CreateManagedAccount (p. 90): The log records whether AWS Control Tower successfully completed every action to create and provision a new account using account factory.
• **UpdateManagedAccount** *(p. 91)*: The log records whether AWS Control Tower successfully completed every action to update a provisioned product that's associated with an account you had previously created by using account factory.

• **EnableGuardrail** *(p. 92)*: The log records whether AWS Control Tower successfully completed every action to enable a guardrail on an OU that was created by AWS Control Tower.

• **DisableGuardrail** *(p. 93)*: The log records whether AWS Control Tower successfully completed every action to disable a guardrail on an OU that was created by AWS Control Tower.

• **SetupLandingZone** *(p. 94)*: The log records whether AWS Control Tower successfully completed every action to set up a landing zone.

• **UpdateLandingZone** *(p. 95)*: The log records whether AWS Control Tower successfully completed every action to update your existing landing zone.

• **RegisterOrganizationalUnit** *(p. 96)*: The log records whether AWS Control Tower successfully completed every action to enable its governance features on an OU.

• **DeregisterOrganizationalUnit** *(p. 97)*: The log records whether AWS Control Tower successfully completed every action to disable its governance features on an OU.

The following sections provide a list of AWS Control Tower lifecycle events, with examples of the details logged for each type of lifecycle event.

## CreateManagedAccount

This lifecycle event records whether AWS Control Tower successfully created and provisioned a new account using account factory. This event corresponds to the AWS Control Tower CreateManagedAccount CloudTrail event. The lifecycle event log includes the `accountName` and `accountId` of the newly-created account, and the `organizationalUnitName` and `organizationalUnitId` of the OU in which the account has been placed.

```json
{
    "version": "0",
    "id": "999cccc9-eaaa-0000-1111-123456789012",
    "detail-type": "AWS Service Event via CloudTrail",
    "source": "aws.controltower",
    "account": "XXXXXXXXXXXX",                                   // Master account ID.
    "region": "us-east-1",                                       // AWS Control Tower home region.
    "resources": [],
    "detail": {
        "eventVersion": "1.05",
        "userIdentity": {
            "accountId": "XXXXXXXXXXXX",
            "invokedBy": "AWS Internal"
        },
        "eventSource": "controltower.amazonaws.com",
        "eventName": "CreateManagedAccount",
        "awsRegion": "us-east-1",
        "sourceIPaddress": "AWS Internal",
        "userAgent": "AWS Internal",
        "eventID": "0000000-0000-0000-1111-123456789012",
        "readOnly": false,
        "eventType": "AwsServiceEvent",
        "serviceEventDetails": {
            "createManagedAccountStatus": {
                "organizationalUnit": {
                    "organizationalUnitId": "XXXXXXXXXXXXX",
                    "organizationalUnitName": "My OU"
                }
            }
        }
    }
}
```
UpdateManagedAccount

This lifecycle event records whether AWS Control Tower successfully updated the provisioned product associated with an account that was created previously by using account factory. This event corresponds to the AWS Control Tower UpdateManagedAccount CloudTrail event. The lifecycle event log includes the accountName and accountId of the associated account, and the organizationalUnitName and organizationalUnitId of the OU in which the updated account is placed.
EnableGuardrail

This lifecycle event records whether AWS Control Tower successfully enabled a guardrail on an OU that is being managed by AWS Control Tower. This event corresponds to the AWS Control Tower `EnableGuardrail` CloudTrail event. The lifecycle event log includes the `guardrailId` and `guardrailBehavior` of the guardrail, and the `organizationalUnitName` and `organizationalUnitId` of the OU on which the guardrail is enabled.

```json
{
    "version": "0",
    "id": "999cccaa-eaaa-0000-1111-123456789012",
    "detail-type": "AWS Service Event via CloudTrail",
    "source": "aws.controltower",
    "account": "XXXXXXXXXXXX",
    "time": "2018-08-30T21:42:18Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "eventVersion": "1.05",
        "userIdentity": {
            "accountId": "XXXXXXXXXXXX",
            "invokedBy": "AWS Internal"
        },
        "eventTime": "2018-08-30T21:42:18Z",
        "eventSource": "controltower.amazonaws.com",
        "eventName": "EnableGuardrail",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "AWS Internal",
        "userAgent": "AWS Internal",
        "eventID": "0000000-0000-0000-1111-123456789012",
        "readOnly": false,
        "eventType": "AwsServiceEvent",
        "serviceEventDetails": {
            "enableGuardrailStatus": {
                "organizationalUnits": [
                    {
                        "organizationalUnitName": "Custom",
                        "organizationalUnitId": "ou-vwxy-18vy4yro"
                    }
                ],
                "guardrails": [
                    {
                        "guardrailId": "AWS-GR_RDS_INSTANCE_PUBLIC_ACCESS_CHECK",
                        "guardrailBehavior": "DETECTIVE"
                    }
                ],
                "state": "SUCCEEDED",
                "message": "AWS Control Tower successfully enabled a guardrail on an organizational unit."
            },
            "requestTimestamp": "2019-11-12T09:07+0000"
        }
    }
}
```
DisableGuardrail

This lifecycle event records whether AWS Control Tower successfully disabled a guardrail on an OU that is being managed by AWS Control Tower. This event corresponds to the AWS Control Tower DisableGuardrail CloudTrail event. The lifecycle event log includes the guardrailId and guardrailBehavior of the guardrail, and the organizationalUnitName and organizationalUnitId of the OU on which the guardrail is disabled.

```json
{
    "version": "0",
    "id": "999cccaa-eaaa-0000-1111-123456789012",
    "detail-type": "AWS Service Event via CloudTrail",
    "source": "aws.controltower",
    "account": "XXXXXXXXXXXX",
    "time": "2018-08-30T21:42:18Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "eventVersion": "1.05",
        "userIdentity": {
            "accountId": "XXXXXXXXXXXX",
            "invokedBy": "AWS Internal"
        },
        "eventTime": "2018-08-30T21:42:18Z",
        "eventSource": "controltower.amazonaws.com",
        "eventName": "DisableGuardrail",
        "awsRegion": "us-east-1",
        "sourceIPAddress": "AWS Internal",
        "userAgent": "AWS Internal",
        "eventID": "0000000-0000-0000-1111-123456789012",
        "readOnly": false,
        "eventType": "AwsServiceEvent",
        "serviceEventDetails": {
            "disableGuardrailStatus": {
                "organizationalUnits": [
                    {
                        "organizationalUnitName": "Custom",
                        "organizationalUnitId": "ou-vwxy-18vy4yro"
                    }
                ],
                "guardrails": [
                    {
                        "guardrailId": "AWS-GR_RDS_INSTANCE_PUBLIC_ACCESS_CHECK",
                        "guardrailBehavior": "DETECTIVE"
                    }
                ],
                "state": "SUCCEEDED",
                "message": "AWS Control Tower successfully disabled a guardrail on an organizational unit."
            },
            "requestTimestamp": "2019-11-12T09:07+0000",
            "completedTimestamp": "2019-11-12T09:01:54+0000"
        }
    }
}
```
SetupLandingZone

This lifecycle event records whether AWS Control Tower successfully set up a landing zone. This event corresponds to the AWS Control Tower SetupLandingZone CloudTrail event. The lifecycle event log includes the rootOrganizationalId, which is ID of the organization that AWS Control Tower creates from the master account. The log entry also includes the organizationalUnitName and organizationalUnitId for each of the OUs, and the accountName and accountId for each account, that are created when AWS Control Tower sets up the landing zone.

```json
{
  "version": "0",
  "id": "999cccaaeaaa-0000-1111-123456789012", // Request ID.
  "detail-type": "AWS Service Event via CloudTrail",
  "source": "aws.controltower",
  "account": "XXXXXXXXXXXX", // Master account ID.
  "region": "us-east-1", // Master account region.
  "resources": [],
  "detail": {
    "eventVersion": "1.05",
    "userIdentity": {
      "accountId": "XXXXXXXXXXXX", // Master-account ID.
      "invokedBy": "AWS Internal"
    },
    "eventSource": "controltower.amazonaws.com",
    "eventName": "SetupLandingZone",
    "awsRegion": "us-east-1", // AWS Control Tower home region.
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "eventID": "CloudTrail_event_ID", // This value is generated by CloudTrail.
    "readOnly": false,
    "eventType": "AwsServiceEvent",
    "serviceEventDetails": {
      "setupLandingZoneStatus": {
        "state": "SUCCEEDED", // Status of entire lifecycle operation.
        "rootOrganizationalId": "r-1234"
      }
    }
  }[
  "organizationalUnits": [
    {
      "organizationalUnitName": "Core", // Core OU name.
      "organizationalUnitId": "ou-adpf-302pk332" // Core OU ID.
    },
    {
      "organizationalUnitName": "Custom", // Custom OU name.
      "organizationalUnitId": "ou-adpf-302pk332" // Custom OU ID.
    }
  ],
  "accounts": [
    // All created accounts are here. Use a list of "account" objects.
  ]
}
```
UpdateLandingZone

This lifecycle event records whether AWS Control Tower successfully updated your existing landing zone. This event corresponds to the AWS Control Tower UpdateLandingZone CloudTrail event. The lifecycle event log includes the rootOrganizationalId, which is ID of the (updated) organization governed by AWS Control Tower. The log entry also includes the organizationalUnitName and organizationalUnitId for each of the OUs, and the accountName and accountId for each account, that was created previously, when AWS Control Tower originally set up the landing zone.
AWS Control Tower User Guide
Lifecycle Events

```
"rootOrganizationalId": "r-1234"
"organizationalUnits": [
    {
        "organizationalUnitName": "Core",
        "organizationalUnitId": "ou-adpf-302pk332"
    },
    {
        "organizationalUnitName": "Custom",
        "organizationalUnitId": "ou-adpf-302pk332"
    }
],
"accounts": [
    {
        "accountName": "Audit",
        "accountId": "XXXXXXXXXXXX"
    },
    {
        "accountName": "Log archive",
        "accountId": "XXXXXXXXXXXX"
    }
],
"requestedTimestamp": "2018-08-30T21:42:18Z",
"completedTimestamp": "2018-08-30T21:42:18Z"
}
```

RegisterOrganizationalUnit

This lifecycle event records whether AWS Control Tower successfully enabled its governance features on an OU. This event corresponds to the AWS Control Tower RegisterOrganizationalUnit CloudTrail event. The lifecycle event log includes the organizationalUnitName and organizationalUnitId of the OU that AWS Control Tower has brought under its governance.

```
{
    "version": "0",
    "id": "999cccaa-eaaa-0000-1111-123456789012",
    "detail-type": "AWS Service Event via CloudTrail",
    "source": "aws.controltower",
    "account": "123456789012",
    "time": "2018-08-30T21:42:18Z",
    "region": "us-east-1",
    "resources": [],
    "detail": {
        "eventVersion": "1.05",
        "userIdentity": {
            "accountId": "XXXXXXXXXXXX",
            "invokedBy": "AWS Internal"
        },
        "eventTime": "2018-08-30T21:42:18Z",
        "eventSource": "controltower.amazonaws.com",
        "eventName": "RegisterOrganizationalUnit",
        "awsRegion": "us-east-1",
        "source IPAddress": "AWS Internal",
        "userAgent": "AWS Internal",
        "eventID": "0000000-0000-0000-1111-123456789012",
        "readOnly": false,
        "eventType": "AwsServiceEvent"
    }
}
```
DeregisterOrganizationalUnit

This lifecycle event records whether AWS Control Tower successfully disabled its governance features on an OU. This event corresponds to the AWS Control Tower DeregisterOrganizationalUnit CloudTrail event. The lifecycle event log includes the organizationalUnitName and organizationalUnitId of the OU on which AWS Control Tower has disabled its governance features.
Compliance Validation for AWS Control Tower

AWS Control Tower is a well-architected service that can help your organization meet your compliance needs with guardrails and best practices. Additionally, third-party auditors assess the security and compliance of a number of the services you can use in your landing zone as a part of multiple AWS compliance programs. These include SOC, PCI, FedRAMP, HIPAA, and others.

For a list of AWS services in scope of specific compliance programs, see AWS Services in Scope by Compliance Program. For general information, see AWS Compliance Programs.

You can download third-party audit reports using AWS Artifact. For more information, see Downloading Reports in AWS Artifact in the AWS Artifact User Guide.

Your compliance responsibility when using AWS Control Tower is determined by the sensitivity of your data, your company’s compliance objectives, and applicable laws and regulations. AWS provides the following resources to help with compliance:

- Security and Compliance Quick Start Guides – These deployment guides discuss architectural considerations and provide steps for deploying security- and compliance-focused baseline environments on AWS.
- Architecting for HIPAA Security and Compliance Whitepaper – This whitepaper describes how companies can use AWS to create HIPAA-compliant applications.
- AWS Compliance Resources – This collection of workbooks and guides might apply to your industry and location.
- AWS Config – This AWS service assesses how well your resource configurations comply with internal practices, industry guidelines, and regulations.
- AWS Security Hub – This AWS service provides a comprehensive view of your security state within AWS that helps you check your compliance with security industry standards and best practices.

Resilience in AWS Control Tower

The AWS global infrastructure is built around AWS Regions and Availability Zones.

AWS Regions provide multiple physically separated and isolated Availability Zones, which are connected by means of low-latency, high-throughput, and highly redundant networking. Availability Zones allow you to design and operate applications and databases that automatically fail over between Availability Zones without interruption. Availability Zones are more highly available, fault tolerant, and scalable than traditional single or multiple data center infrastructures.

AWS Control Tower is available in these AWS Regions:

- US East (N. Virginia)
Infrastructure Security

Your *home region* is defined as the AWS Region in which your landing zone was set up.

For more information about AWS Regions and Availability Zones, see [AWS Global Infrastructure](https://aws.amazon.com/about-aws/global-infrastructure/).

**Infrastructure Security in AWS Control Tower**

AWS Control Tower is protected by the AWS global network security procedures that are described in the *Amazon Web Services: Overview of Security Processes* whitepaper.

You use AWS published API calls for access to AWS services and resources within your landing zone through the network. Clients must support Transport Layer Security (TLS) 1.0 or later. We recommend TLS 1.2 or later. Clients must also support cipher suites with perfect forward secrecy (PFS) such as Ephemeral Diffie-Hellman (DHE) or Elliptic Curve Ephemeral Diffie-Hellman (ECDHE). Most modern systems such as Java 7 and later support these modes.

Additionally, requests must be signed by using an access key ID and a secret access key that is associated with an IAM principal. Or you can use the *AWS Security Token Service* (AWS STS) to generate temporary security credentials to sign requests.

You can set up security groups to provide additional network infrastructure security for your AWS Control Tower landing zone workloads. For more information, see [Walkthrough: Setting Up Security Groups in AWS Control Tower With AWS Firewall Manager (p. 112)](https://aws.amazon.com/documentation/awsclearinghouse/latest/guides/creating-security-groups/).

**Configuration Update Management in AWS Control Tower**

It is the responsibility of the members of your central cloud administrators’ team to keep your landing zone updated. Updating your landing zone ensures that AWS Control Tower is patched and updated. In addition, to protect your landing zone from potential compliance issues, the members of the central cloud administrator team should resolve drift issues as soon as they’re detected and reported.

**Note**

The AWS Control Tower console indicates when your landing zone needs to be updated. If you don’t see an option to update, your landing zone is already up to date.

**About Updates**

Updates are required to correct governance drift, or to move to a new version of AWS Control Tower. To perform a complete update of AWS Control Tower, you must update your landing zone first and then update the enrolled accounts individually. You may need to perform three types of updates at different times.

- **A landing zone update:** Most often this type of update is performed by choosing *Update* on the *Settings* page. You may need to perform a landing zone update to repair certain types of drift, and you can choose *Repair* when necessary.
• **An update of one or more individual accounts**: You must update accounts if the associated information changes, or if certain types of drift have occurred. Accounts may be updated by following the manual process, or with an automated approach. Both are described in later sections of this page.

• **A full update**: A full update includes an update of your landing zone, followed by an update of all the enrolled accounts in your registered OU. Full updates are required with a new release of AWS Control Tower such as 2.3, 2.4, and so forth.

---

**Update Your Landing Zone**

The easiest way to update your AWS Control Tower landing zone is through the **Settings** page. Navigate to the **Settings** page by choosing **Settings** in the left navigation.

The **Settings** page shows you the current version of your landing zone, and it lists any updated versions that may be available. You can choose the **Update** button if you need to update your version. If the **Update** button appears greyed-out, you do not need to update.

**Note**
Alternatively, you can update your landing zone manually. The update takes approximately the same amount of time, whether you use the **Update** button or the manual process. To perform a manual update of your landing zone only, see steps 1 and 2 that follow.

The following procedure walks you through the steps of a full update for AWS Control Tower manually. To update an individual account, start at Step 3.

**To update your landing zone manually**

2. Review the information in the wizard and choose **Update**. This updates the backend of the landing zone as well as your shared accounts. This process can take a little more than an hour.
3. Update your member accounts. From the navigation pane, choose **Accounts**.
4. Choose **Enroll account** to open the AWS Service Catalog console and the Account Factory product.
5. From the navigation pane, choose **Provisioned products list**.
6. For each account listed, perform the following steps to update all your member accounts:
    a. From the menu for the account, choose **Provisioned product details**.
    b. Make a note of the following parameters:
       - **SSOUserEmail** (Available in provisioned product details)
       - **AccountEmail** (Available in provisioned product details)
       - **SSOUserFirstName** (Available in SSO)
       - **SSOUserLastName** (Available in SSO)
       - **AccountName** (Available in SSO)
    c. From **ACTIONS**, choose **Update**.
    d. Choose the radio button next to the **Version** of the product you want to update, and choose **NEXT**.
    e. Provide the parameter values that were mentioned previously. For **ManagedOrganizationalUnit** choose the OU that the account is in. You can find this information in the AWS Control Tower console, under **Accounts**.
    f. Choose **NEXT**.
    g. Review your changes, and then choose **UPDATE**. This process can take a few minutes per account.
Resolve Drift

When you create your AWS Control Tower landing zone, the landing zone and all the OUs, accounts, and resources are compliant with all of the governance rules enforced by your guardrails, whether mandatory or elective. As you and your organization members use the landing zone, changes in compliance status may occur. Some changes may be accidental, and some may be made intentionally to respond to time-sensitive operational events. Regardless, changes can complicate your compliance story.

Resolving drift helps to ensure your organization's compliance with governance regulations. Drift resolution is a regular operations task for your master account administrators.

Drift detection is automatic in AWS Control Tower. It helps you identify resources that need changes or configuration updates that must be made to resolve the drift.

To repair most types of drift, choose Repair on the Settings page. The Repair button becomes selectable when drift has occurred. For more information, see Detecting and Resolving Drift in AWS Control Tower (p. 71).

Deploying AWS Control Tower to a New AWS Region

This section describes the behavior you can expect when you deploy your AWS Control Tower landing zone into a new AWS Region. Generally, this type of deployment is performed through the Update function of the AWS Control Tower console.

Note
We recommend that you avoid expanding your AWS Control Tower landing zone into AWS Regions in which you do not require your workloads to run.

During deployment into a new AWS Region, AWS Control Tower updates the landing zone, which means that it baselines your landing zone to operate actively in the new Region. Individual accounts within your organizational units (OUs) that are managed by AWS Control Tower are not updated as part of this landing zone update process. Therefore, you must apply updates to your accounts individually.

Certain significant behavioral changes in AWS Control Tower detective guardrails are expected as a result of a deployment to a new AWS Region:

• What exists stays the same. Guardrail behavior, detective as well as preventive, is unchanged for existing accounts, in existing OUs, in existing Regions.
• You can't apply new detective guardrails to existing OUs containing accounts that are not updated. When you've deployed your AWS Control Tower landing zone into the new Region (by updating it), you must update existing accounts in your existing OUs before you can enable new detective guardrails on those OUs and accounts.
• Your existing detective guardrails begin working in the new Region as soon as you update the accounts. When you update your AWS Control Tower landing zone to deploy into the new Region and then update an account, the detective guardrails that already are enabled on the OU will begin working on that account in the new Region.
• Update only to Regions in which you need to run. Due to this behavior, it can be time-consuming to deploy your AWS Control Tower landing zone into new AWS Regions if your OUs manage a lot of member accounts. Therefore, we recommend that you avoid expanding your AWS Control Tower deployment into AWS Regions in which you do not require your workloads to run.

Methods for automated updating of accounts

One method of updating individual accounts after deployment to a new region is by using the API framework of AWS Service Catalog and the AWS CLI to update the accounts in a batch process. You'd call the UpdateProvisionedProduct API of AWS Service Catalog for each account. You can write a script...
to update the accounts, one by one, with this API. More information about this approach is available in a blog post.

You must wait for each account update to succeed before beginning the next account update. Therefore, the process may take a long time if you have a lot of accounts, but it is not complicated. For more information about this approach, see the Walkthrough: Automated Account Provisioning in AWS Control Tower (p. 110).

**Note**

The Video Walkthrough (p. 112) is designed for automated account provisioning, but the steps also apply to account updating, if you remember to call the "UpdateProvisionedProduct" API instead of the "ProvisionProduct" API.

A further step of automation is to check for the AWS Control Tower lifecycle "UpdateLandingZone" **Succeed** event and use it as a trigger to begin updating individual accounts as described in the video. A lifecycle event marks the completion of a sequence of activities, so the occurrence of this event means that a landing zone update is complete. The landing zone update must be complete before account updates begin. For more information about working with lifecycle events, see Lifecycle Events.
Limitations and Quotas in AWS Control Tower

This chapter covers the AWS service limitations and quotas that you should keep in mind as you use AWS Control Tower. If you’re unable to set up your landing zone due to a service quota issue, contact AWS Support.

Limitations in AWS Control Tower

This section describes known limitations and unsupported use cases in AWS Control Tower.

- Nested OUs are not displayed in the AWS Control Tower console.
- Creation of nested OUs from the AWS Control Tower console is not supported.
- Email addresses of shared accounts in the Core OU can be changed, but you must update your landing zone to see these changes in the AWS Control Tower console.
- A limit of 5 SCPs per OU applies to OUs in your AWS Control Tower landing zone.

Quotas for Integrated Services

Each AWS service has its own quotas and limits. You can find the quotas for each service in its documentation. For more information, see the related links:

- **AWS CloudFormation** – [AWS CloudFormation Quotas](#)
- **AWS CloudTrail** – [Quotas in AWS CloudTrail](#)
- **Amazon CloudWatch** – [CloudWatch Quotas](#)
- **AWS Config** – [AWS Config Quotas](#)
- **AWS Identity and Access Management** – [Quotas for IAM Entities and Objects](#)
- **AWS Lambda** – [AWS Lambda Quotas](#)
- **AWS Organizations** – [Quotas for AWS Organizations](#)
- **Amazon Simple Storage Service** – [Bucket Restrictions and Quotas](#)
- **AWS Service Catalog** – [AWS Service Catalog Default Service Quotas](#)
- **AWS Single Sign-On** – [Quotas in AWS SSO](#)
- **Amazon Simple Notification Service** – [Amazon Simple Notification Service (Amazon SNS) Quotas](#)
- **AWS Step Functions** – [Quotas](#)
Walkthroughs

This chapter contains walkthrough procedures that can help you in your use of AWS Control Tower.

Topics

• Walkthrough: Cleaning up AWS Control Tower Managed Resources (p. 104)
• Walkthrough: Configuring AWS Control Tower Without a VPC (p. 108)
• Walkthrough: Customize Your AWS Control Tower Landing Zone (p. 110)
• Walkthrough: Automated Account Provisioning in AWS Control Tower (p. 110)
• Walkthrough: Setting Up Security Groups in AWS Control Tower With AWS Firewall Manager (p. 112)

Walkthrough: Cleaning up AWS Control Tower Managed Resources

When your landing zone was set up, AWS Control Tower provisioned resources and services in your landing zone on your behalf. For example, an AWS Organizations organization with multiple accounts and organizational units (OUs) were provisioned. Additionally, guardrails were deployed in your accounts using AWS CloudFormation stacks, stack sets, and AWS Organizations policies.

Note

If you delete all of your landing zone resources manually, it is not the same as decommissioning the landing zone. Complete decommissioning requires assistance from AWS Support.

Manual Cleanup of AWS Control Tower Resources

The following procedures guide you through manual methods of cleaning up AWS Control Tower resources. These procedures can be followed any time you need to delete resources from you landing zone. Two types of tasks may require cleanup of resources:

• To delete resources as you manage your landing zone in ordinary situations.
• To clean up resources that remain after automated decommissioning, in cooperation with AWS Support.

Before performing these procedures, unless it's otherwise indicated, you must be signed in to the AWS Management Console in the home Region for your landing zone, and you must be signed in as an IAM user with administrative permissions for the master account that contains your landing zone.

Warning

These are destructive actions that can introduce governance drift into your AWS Control Tower setup. They cannot be undone.

Topics

• Delete SCPs (p. 105)
• Delete StackSets and Stacks (p. 105)
• Delete Amazon S3 Buckets in the Log Archive Account (p. 106)
Delete SCPs

AWS Control Tower uses service control policies (SCPs) for its guardrails. This procedure walks through how to delete the SCPs specifically related to AWS Control Tower.

To delete AWS Organizations SCPs

1. Open the Organizations console at https://console.aws.amazon.com/organizations/.
2. Open the Policies tab, and find the Service Control Policies (SCPs) that have the prefix aws-guardrails- and do the following for each SCP:
   a. Detach the SCP from the associated OU.
   b. Delete the SCP.

Delete StackSets and Stacks

AWS Control Tower uses StackSets and stacks to deploy AWS Config Rules related to guardrails in your landing zone. The following procedures walk through how to delete these specific resources.

To delete AWS CloudFormation StackSets

2. From the left navigation menu, choose StackSets.
3. For each StackSet with the prefix AWSControlTower, do the following. If you have many accounts in a StackSet, this can take some time.
   a. Choose the specific StackSet from the table in the dashboard. This opens the properties page for that StackSet.
   b. At the bottom of the page, in the Stacks table, make a record of the AWS account IDs for all the accounts in the table. Copy the list of all accounts.
   c. Choose Manage StackSet to open the management wizard.
   d. From Select action, choose Delete stacks, and choose Next.
   e. On Set deployment options, from Specify accounts, choose Delete stacks from account.
   f. In the text field, enter the AWS account IDs you made a record of in step 3.b, separated by commas. For example: 123456789012, 098765431098, and so on.
   g. From Specify regions, choose Add all, leave the rest of the parameters on the page set to their defaults, and choose Next.
   h. On the Review page, review your choices, and then choose Delete stacks.
   i. On the StackSet properties page, you can begin this procedure again for your other StackSets.
4. The process is complete when the records in the Stacks table of the different StackSets properties pages are empty.
5. When the records in the Stacks table are empty, choose Delete StackSet.

To delete AWS CloudFormation stacks

2. From the Stacks dashboard, search for all of the stacks with the prefix AWSControlTower.
3. For each stack in the table, do the following:
   a. Choose the check box next to the name of the stack.
   b. From the Actions menu, choose Delete Stack.
   c. In the dialog box that opens, review the information to make sure it’s accurate, and choose Yes, Delete.

Delete Amazon S3 Buckets in the Log Archive Account

The following procedures guide you through how to sign in to the log archive account as an AWS SSO user in the AWSControlTowerExecution group and then delete the Amazon S3 buckets in your log archive account.

To sign in to your log archive account with the right permissions
1. Open the Organizations console at https://console.aws.amazon.com/organizations/.
2. From the Accounts tab, find the Log archive account.
3. From the right pane that opens, make a record of the log archive account number.
4. From the navigation bar, choose your account name to open your account menu.
5. Choose Switch Role.
6. On the page that opens, provide the account number for the log archive account in Account.
7. For Role, enter AWSControlTowerExecution.
8. The Display Name populates with text.
9. Choose your favorite Color.
10. Choose Switch Role.

To delete Amazon S3 buckets
1. Open the Amazon S3 console at https://console.aws.amazon.com/s3/.
2. Search for bucket names that contain aws-controltower.
3. For each bucket in the table, do the following:
   a. Choose the check box for the bucket in the table.
   b. Choose Delete.
   c. In the dialog box that opens, review the information to make sure it’s accurate, enter the name of the bucket to confirm, and then choose Confirm.

Clean Up Account Factory

The following procedure guides you through how to sign in as an AWS SSO user in the AWSServiceCatalogAdmins group and then clean up your Account Factory accounts.

To sign in to your master account with the right permissions
1. Go to your user portal URL at directory-id.awsapps.com/start
2. From AWS Account, find the Master account.
3. From AWSServiceCatalogAdminFullAccess, choose Management console to sign in to the AWS Management Console as this role.

To clean up Account Factory

1. Open the AWS Service Catalog console at https://console.aws.amazon.com/servicecatalog/.
2. From the left navigation menu, choose Portfolios list.
3. In the Local Portfolios table, search for a portfolio named AWS Control Tower Account Factory Portfolio.
4. Choose the name of that portfolio to go to its details page.
5. Expand the Constraints section of the page, and choose the radio button for the constraint with the product name AWS Control Tower Account Factory.
6. Choose REMOVE CONSTRAINTS.
7. In the dialog box that opens, review the information to make sure it's accurate, and then choose CONTINUE.
8. From the Products section of the page, choose the radio button for the product named AWS Control Tower Account Factory.
9. Choose REMOVE PRODUCT.
10. In the dialog box that opens, review the information to make sure it's accurate, and then choose CONTINUE.
11. Expand the Users, Groups, and Roles section of the page, and choose the check boxes for all the records in this table.
12. Choose REMOVE USERS, GROUP OR ROLE.
13. In the dialog box that opens, review the information to make sure it's accurate, and then choose CONTINUE.
14. From the left navigation menu, choose Portfolios list.
15. In the Local Portfolios table, search for a portfolio named AWS Control Tower Account Factory Portfolio.
16. Choose the radio button for that portfolio, and then choose DELETE PORTFOLIO.
17. In the dialog box that opens, review the information to make sure it's accurate, and then choose CONTINUE.
18. From the left navigation menu, choose Product list.
20. Choose the product to open the Admin product details page.
21. From Actions, choose Delete product.
22. In the dialog box that opens, review the information to make sure it's accurate, and then choose CONTINUE.

Clean Up Roles and Policies

These procedures walk you through how to clean up the roles and policies that were created when your landing zone was set up.

To delete the AWS SSO AWSServiceCatalogEndUserAccess role

2. Change your AWS Region to US East (N. Virginia).
3. From the left navigation menu, choose **AWS accounts**.
4. Choose your master account link.
5. Choose the dropdown for **Permission sets**, select **AWSServiceCatalogEndUserAccess**, and then choose **Remove**.
6. Choose **AWS accounts** from the left panel.
7. Open the **Permission sets** tab.
8. Select **AWSServiceCatalogEndUserAccess** and delete it.

**To delete IAM roles**
2. From the left navigation menu, choose **Roles**.
3. From the table, search for roles with the name **AWSControlTower**.
4. For each role in the table, do the following:
   a. Choose the check box for the role.
   b. Choose **Delete role**.
   c. In the dialog box that opens, review the information to make sure it's accurate, and then choose **Yes, delete**.

**To delete IAM policies**
2. From the left navigation menu, choose **Policies**.
3. From the table, search for policies with the name **AWSControlTower**.
4. For each policy in the table, do the following:
   a. Choose the check box for the policy.
   b. Choose **Policy actions**, and **Delete** from the dropdown menu.
   c. In the dialog box that opens, review the information to make sure it's accurate, and then choose **Delete**.

**AWS Control Tower Clean Up Help**

If you encounter any issues that you can't resolve during this clean up process, contact **AWS Support**.

**Walkthrough: Configuring AWS Control Tower Without a VPC**

This topic walks through how to configure your AWS Control Tower accounts without a VPC.

If your workload does not require a VPC, you can do the following:

- You can delete the AWS Control Tower master account virtual private cloud (VPC). This VPC was created when you set up your landing zone.
- You can change your Account Factory settings so that new AWS Control Tower accounts are created without an associated VPC.
Delete the AWS Control Tower Master Account VPC

Outside of AWS Control Tower, every AWS customer has a default VPC, which you can view on the Amazon Virtual Private Cloud (Amazon VPC) console at https://console.aws.amazon.com/vpc/. You'll recognize the default VPC, because its name always includes the word *(default)* at the end of the name.

When you set up a AWS Control Tower landing zone, AWS Control Tower deletes your AWS default VPC and creates a new AWS Control Tower default VPC. The new VPC is associated with your AWS Control Tower master account. This topic refers to that new VPC as the *Control Tower master account VPC*.

When you view your AWS Control Tower master account VPC in the Amazon VPC console, you will not see the word *(default)* at the end of the name. If you have more than one VPC, you must use the assigned CIDR range to identify the correct AWS Control Tower master account VPC.

You can delete the AWS Control Tower master account VPC, but if you later need a VPC in AWS Control Tower, you must create it yourself.

**To delete the AWS Control Tower master account VPC**

1. Open the Amazon VPC console at https://console.aws.amazon.com/vpc/.
2. Search for VPC or select VPC from the AWS Service Catalog options. You then see the VPC Dashboard.
3. From the menu on the left, choose Your VPCs. You then see a list of all your VPCs.
4. Identify the AWS Control Tower master account VPC by its CIDR range.
5. To delete the VPC, choose Actions and then choose Delete VPC.

An AWS *(default)* VPC already exists in every region for the AWS Control Tower master account. To follow security best practices, if you choose to delete the AWS Control Tower master account VPC, it's best also to delete the AWS default VPC associated with the master account from all AWS Regions. Therefore, to secure the master account, remove the default VPC from each Region, as well as removing the VPC created by Control Tower in your AWS Control Tower home region.

Create an Account in AWS Control Tower Without a VPC

If your end user workloads do not require VPCs, you can use this method to set up user accounts that don't have VPCs created for them automatically.

From the AWS Control Tower dashboard, you can view and edit your network configurations settings. After you change the settings so that AWS Control Tower accounts are created without an associated VPC, all new accounts are created without a VPC until you change the settings again.

**To configure Account Factory for creating accounts without VPCs**

2. Choose Account Factory from the menu on the left.
3. You then see the Account Factory page with the Network Configuration section.
4. Note the current settings if you intend to restore them later.
5. Choose the Edit button in the Network Configuration section.
6. In the Edit account factory network configuration page, go to the VPC Configuration options for new accounts section.
   a. Turn off the Internet-accessible subnet toggle switch.
Walkthrough: Customize Your AWS Control Tower Landing Zone

You can add customizations to your AWS Control Tower landing zone using an AWS CloudFormation template and service control policies (SCPs). You can deploy the custom template and policies to individual accounts and organizational units (OUs) within your organization.

This solution integrates with AWS Control Tower lifecycle events to ensure that resource deployments stay in sync with the landing zone. For example, when a new account is created using the AWS Control Tower account factory, the solution ensures that all resources attached to the account’s OUs are deployed automatically.

The deployment documentation for this AWS Control Tower solution architecture is available through the AWS Solutions web page.

Walkthrough: Automated Account Provisioning in AWS Control Tower

AWS Control Tower is integrated with several other AWS services, such as AWS Service Catalog. You can use the APIs to create and provision your member accounts in AWS Control Tower.

The video shows you how to provision accounts in an automated, batch fashion, by calling the AWS Service Catalog APIs. For provisioning, you’ll call the ProvisionProduct API from the AWS command line interface (CLI), and you’ll specify a JSON file that contains the parameters for each account you’d like to set up. The video illustrates installing and using the AWS Cloud9 development environment to perform this work.

**Note**

You also can use this approach for automating account updates, by calling the UpdateProvisionedProduct API of AWS Service Catalog for each account. You can write a script to update the accounts, one by one.

Here is a sample template you can use to help configure your automation administration role in the master account.
Here is a sample template you can use to help you set up your automation execution role.

```yaml
AWSTemplateFormatVersion: "2010-09-09"
Description: "Create automation execution role for creating Sample Additional Role."

Parameters:
  AdminAccountId:
    Type: "String"
    Description: "Account ID for the administrator account (typically master, security or shared services)."
  AdminRoleName:
    Type: "String"
    Description: "Role name for automation administrator access."
  ExecutionRoleName:
    Type: "String"
    Description: "Role name for automation execution."
  SessionDurationInSecs:
    Type: "Number"
    Description: "Maximum session duration in seconds."
    Default: 14400

Resources:
  # This needs to run after AdminRoleName exists.
  ExecutionRole:
    Type: "AWS::IAM::Role"
    Properties:
      RoleName: !Ref ExecutionRoleName
      MaxSessionDuration: !Ref SessionDurationInSecs
      AssumeRolePolicyDocument:
        Version: "2012-10-17"
```
Video Walkthrough

This video (7:08) describes how to automate account deployments in AWS Control Tower. For better viewing, select the icon at the lower right corner of the video to enlarge it to full screen. Captioning is available.

Video Walkthrough of Automated Account Provisioning in AWS Control Tower.

Walkthrough: Setting Up Security Groups in AWS Control Tower With AWS Firewall Manager

The video shows you how to use the AWS Firewall Manager service to provide improvements to your network security for AWS Control Tower. You can designate a security administrator account that's enabled to set up security groups. You will see how you can configure security policies and enforce security rules for your AWS Control Tower organizations, and how you can remediate non-compliant resources by applying policies automatically. You can view the security groups that are in effect for each account and resource (such as an EC2 instance) in your organization.

You can create your own firewall policies, or you can subscribe to rules from trusted vendors.

Set Up Security Groups With AWS Firewall Manager

This video (8:02) describes how to set up better network infrastructure security for your resources and workloads in AWS Control Tower. For better viewing, select the icon at the lower right corner of the video to enlarge it to full screen. Captioning is available.

Video Walkthrough of Firewall Setup in AWS Control Tower.

For more information, see the documentation on how to set up AWS WAF.
Enable AWS Control Tower on an Existing Organization

You can add AWS Control Tower governance to an existing organization by setting up a landing zone (LZ) as outlined in the AWS Control Tower User Guide at Getting Started, Step 2. Here's what to expect when you set up your AWS Control Tower landing zone in an existing organization.

- You can have one landing zone per AWS Organizations organization.
- AWS Control Tower uses the master account from your existing AWS Organizations organization as its master account. No new master account is needed. However, AWS Control Tower sets up two new accounts in a registered OU: an audit account and a logging account.
- Your organization's service limits must allow for the creation of these two additional accounts.
- Once you've launched your landing zone in the registered OU, AWS Control Tower guardrails apply automatically to all accounts in that OU.

Here's more detail about how AWS Control Tower guardrails do not apply to your OUs in AWS Organizations that don't have landing zones set up:

- New accounts created outside of AWS Control Tower Account Factory are not bound by the registered OU's guardrails.
- New accounts created in OUs that are not registered with AWS Control Tower are not bound by guardrails, unless you specifically Enroll those accounts into AWS Control Tower. See Enrolling an Existing AWS Account in AWS Control Tower (p. 114) for more information about enrolling accounts.
- Additional existing AWS Organizations, existing accounts, and any new OUs or any accounts that you create outside of AWS Control Tower, are not bound by guardrails.

For an overview of the process of setting up an AWS Control Tower landing zone in your existing organization, see the video in the next section.

Enable a Landing Zone in Existing AWS Organizations

This video (6:52) describes how to set up and enable an AWS Control Tower landing zone in existing AWS Organizations. For better viewing, select the icon at the lower right corner of the video to enlarge it to full screen. Captioning is available.

Enable AWS Control Tower for Existing Organizations
Enrolling an Existing AWS Account in AWS Control Tower

You can extend AWS Control Tower governance to individual, existing AWS accounts by enrolling them into an AWS Control Tower organizational unit (OU). Eligible accounts exist in unregistered OUs that are part of the same AWS Organizations organization as the AWS Control Tower OU.

Before you can enroll an existing AWS account into AWS Control Tower you must give permission for AWS Control Tower to manage, or govern, the account. Specifically, AWS Control Tower requires permission to establish trusted access between AWS CloudFormation and AWS Organizations on your behalf, so that AWS CloudFormation can deploy your stack automatically to the accounts in your selected organization.

To learn more about trusted access and AWS CloudFormation StackSets, see AWS CloudFormation StackSets and AWS Organizations. When trusted access is enabled, AWS CloudFormation can create, update, or delete stacks across multiple accounts and AWS Regions with a single operation. AWS Control Tower relies on this trust capability so it can apply roles and permissions to existing accounts before it moves them into a registered organizational unit and thereby brings them under governance.

During the enrollment process, AWS Control Tower performs these actions:

- Baselines the account, which includes deploying these stack sets:
  - AWSControlTowerBP-BASELINE-CLOUDTRAIL
  - AWSControlTowerBP-BASELINE-CLOUDWATCH
  - AWSControlTowerBP-BASELINE-CONFIG
  - AWSControlTowerBP-BASELINE-ROLES
  - AWSControlTowerBP-BASELINE-SERVICE-ROLES
  It is a good idea to review the templates of these stack sets and make sure that they don't conflict with your existing policies.
- Identifies the account through AWS Single Sign-On or AWS Organizations.
- Places the account into the OU that you've specified. Be sure to apply all SCPs that are applied in the current OU, so that your security posture remains consistent.
- Applies mandatory guardrails to the account by means of the SCPs that apply to the selected OU as a whole.
- Adds the AWS Config rules that apply the AWS Control Tower detective guardrails to the account.

**Enrolling Existing Accounts With VPCs**

AWS Control Tower handles VPCs differently when you provision a new account in Account Factory than when you enroll an existing account.

- When you create a new account, AWS Control Tower automatically removes the AWS default VPC and creates a new VPC for that account.
- When you enroll an existing account, AWS Control Tower does not create a new VPC for that account.
- When you enroll an existing account, AWS Control Tower does not remove any existing VPC or AWS default VPC associated with the account.
Recommended: You can set up a two-step approach to account enrollment

- First, use an AWS Config conformance pack to evaluate how your accounts may be affected by some AWS Control Tower guardrails. To determine how enrollment into AWS Control Tower may affect your accounts, see AWS Control Tower Detective Guardrails as an AWS Config Conformance Pack.
- Next, you may wish to enroll the account. If the compliance results are satisfactory, the migration path is easier because you can enroll the account without unexpected consequences.
- After you've done your evaluation, if you decide to set up an AWS Control Tower landing zone, you may need to remove the AWS Config delivery channel and configuration recorder that were created for your evaluation. Then you'll be able to set up AWS Control Tower successfully.

**Note**
The conformance pack also works in situations where the accounts are located in OUs registered by AWS Control Tower, but the workloads run within AWS Regions that don't have AWS Control Tower support. You can use the conformance pack to manage resources in accounts that exist in Regions where AWS Control Tower is not deployed.

---

**Prerequisites for Enrollment**

Before you can enroll an existing account in AWS Control Tower, the account must have the following roles, permissions, and trust relationships in place.

**Role Name:** AWSControlTowerExecution

**Role Permission:** AdministratorAccess (AWS managed policy)

**Role Trust Relationship:**

```json
{
   "Version": "2012-10-17",
   "Statement": [
   {
      "Effect": "Allow",
      "Principal": {
         "AWS": "arn:aws:iam::Master Account ID:root"
      },
      "Action": "sts:AssumeRole",
      "Condition": {}
   }
   ]
}
```

To check other prerequisites for enrollment, see Getting Started with AWS Control Tower.

After the AdministratorAccess permission is in place in your existing account, follow these steps to enroll the account:

**To enroll an individual account in AWS Control Tower**

- Navigate to the AWS Control Tower Account Factory page and select Enroll account.
- Specify the current email address of the existing account you'd like to enroll in AWS Control Tower.
- Specify the first and last name of the account owner.
• Specify the organizational unit (OU) in which you'd like to enroll the account.
• Choose Enroll account.

Common Causes for Failure of Enrollment

• Your IAM principal may lack the necessary permissions to provision an account. To enroll an existing account, the AWSControlTowerExecution role must be present in the account you're enrolling.
• AWS Security Token Service (AWS STS) is disabled in your AWS account in your home region, or in any region supported by AWS Control Tower.
• You may be signed in to an account that needs to be added to the Account Factory Portfolio in AWS Service Catalog. The account must be added before you'll have access to Account Factory so you can create or enroll an account in AWS Control Tower. If the appropriate user or role is not added to the Account Factory Portfolio, you'll receive an error when you attempt to add an account.
• You may be signed in as root.
• The account you're trying to enroll may have AWS Config settings that are residual. In particular, the account must not have a configuration recorder or delivery channel, so these must be deleted through the AWS CLI before you can enroll an account.

For more information about how AWS Control Tower works with roles when you're creating new accounts or enrolling existing accounts, see How AWS Control Tower Works With Roles to Create and Manage Accounts (p. 9).

Important
The account that you wish to enroll must exist in the same AWS Organizations organization as the AWS Control Tower master account. The account that exists can be enrolled only into the same organization as the AWS Control Tower master account, in an OU that already is registered with AWS Control Tower. Otherwise, enrollment will fail.

Automated Enrollment of AWS Organizations Accounts

You can use the enrollment method described in a blog post called Enroll existing AWS accounts into AWS Control Tower to enroll your AWS Organizations accounts into AWS Control Tower with a programmatic process.
Decommissioning a Landing Zone

AWS Control Tower allows you to set up and govern secure multi-account AWS environments, known as landing zones. The process of cleaning up all of the resources allocated by AWS Control Tower is referred to as decomposition a landing zone.

To decommission your landing zone with a process that is mostly automated, contact AWS Support, and they will assist you with further steps you need to follow. For a complete list of actions performed during decommissioning, see Overview of the Decommissioning Process (p. 117).

Your data and your existing AWS Organizations are not changed by the decommissioning process, in the following ways.

- AWS Control Tower does not remove your data, it only removes parts of the landing zone that it created.
- After the decommissioning process is complete, a few resource artifacts remain, such as S3 buckets and Amazon CloudWatch Logs log groups. These resources must be deleted manually before you set up another landing zone, and to avoid possible costs associated with maintaining certain resources.
- You can’t use automated decommissioning to remove a landing zone that’s partially set up. If your landing zone setup process fails, you must resolve the failure state and set it up all the way to make automated decommissioning possible, or you must manually delete the resources individually.

Decommissioning a landing zone is a process with significant consequences, and it cannot be undone. The decommissioning actions taken by AWS Control Tower and the artifacts that remain after decommissioning are described in the following sections.

Important
We strongly recommend that you perform this decommissioning process only if you intend to stop using your landing zone. It is not possible to re-create your existing landing zone after you’ve decommissioned it.

Overview of the Decommissioning Process

When you request decommissioning of your landing zone, AWS Control Tower does the following actions.

- Disables each detective guardrail enabled in the landing zone. AWS Control Tower deletes the AWS CloudFormation resources supporting the guardrail.
- Disables each preventive guardrail by removing service control policies (SCPs) from AWS Organizations. If a policy is empty (which it should be after removing all SCPs managed by AWS Control Tower), AWS Control Tower detaches and deletes the policy entirely.
- Deletes all blueprints deployed as CloudFormation StackSets.
- Deletes all blueprints deployed as CloudFormation Stacks across all Regions.
- For each provisioned account, AWS Control Tower does the following actions during the decommissioning process.
  - Deletes records of each account factory account.
  - Revokes the AWS Control Tower permissions to the account by removing the IAM role that AWS Control Tower created (unless additional policies have been added to it) and recreates the standard OrganizationsFullAccessRole IAM role.
• Removes records of the account from AWS Service Catalog.
• Removes the account factory product and portfolio from AWS Service Catalog.
• Deletes the blueprints for the core (logging and audit) accounts.
• Revokes the AWS Control Tower permissions from the core accounts by removing the IAM role that AWS Control Tower created (unless additional policies have been added to it) and recreates the OrganizationsFullAccessRole IAM role.
• Deletes records related to the core accounts.
• Deletes records related to customer-created OUs.
• Deletes internal records that identify the home Region.

Note
After decommissioning, you may wish to remove the Account Factory VPC blueprint (BP_ACCOUNT_FACTORY_VPC) to clean up the routes and NAT gateways, if your VPC was not empty.

Resources Not Removed During Decommissioning

Decommissioning a landing zone does not fully reverse the AWS Control Tower setup process. Certain resources remain, which may be removed manually.

AWS Organizations

For customers without existing AWS Organizations organizations, AWS Control Tower sets up an organization with two organizational units (OUs), named Core and Custom. When you decommission your landing zone, the hierarchy of the organization is preserved, as follows:

• Organizational Units (OUs) you created from the AWS Control Tower console are not removed.
• The core and custom OUs are not removed.
• The organization is not deleted from AWS Organizations.
• No accounts in AWS Organizations (core, provisioned, or master) are moved or removed.

AWS Single Sign-On (SSO)

For customers without an existing AWS SSO directory, AWS Control Tower sets up AWS SSO and configures an initial directory. When you decommission your landing zone, AWS Control Tower makes no changes to AWS SSO. If needed, you can delete the AWS SSO information stored in your master account manually. In particular, these areas are unchanged by decommissioning:

• Users created with Account Factory are not removed.
• Groups created by AWS Control Tower setup are not removed.
• Permission sets created by AWS Control Tower are not removed.
• Associations between AWS accounts and AWS SSO permission sets are not removed.
• AWS SSO directories are not changed.

Amazon S3 Buckets

During setup, AWS Control Tower creates buckets in the logging account for logging and for logging access. When you decommission your landing zone, the following resources are not removed:
How to Decommission a Landing Zone

After you've contacted AWS Support and received confirmation that they've enabled your account for decommissioning, you'll be able to see a section on your Settings page that allows you to follow the procedure given here. This procedure will not work until AWS Support has enabled your account for decommissioning.

1. Navigate to the Settings page in the AWS Control Tower console.
2. Choose Decommission your landing zone within the Decommission your landing zone section.
3. A dialog appears, explaining the action you are about to perform, with a required confirmation process. To confirm your intent to decommission, you must select every box and type the confirmation as requested.
   
   **Important**
   
   The decommissioning process cannot be undone.

4. If you confirm your intent to decommission your landing zone, you are redirected to the AWS Control Tower home page while decommissioning is in progress. The process may require up to two hours.
5. When decommissioning has succeeded, you must delete remaining resources manually before setting up a new landing zone from the AWS Control Tower console. These remaining resources include some specific S3 buckets, organizations, and CloudWatch Logs log groups.

   **Note**
   *These actions may have significant consequences for your billing and compliance activities. For example, failure to delete these resources can result in unexpected charges.*

   For more information about how to delete resources manually, see Manual Cleanup of AWS Control Tower Resources (p. 104).

Manual Cleanup Tasks Required After Decommissioning

- You must specify different email addresses for the logging and audit accounts if you create a new landing zone after decommissioning one.
- The CloudWatch Logs log group, `aws-controltower/CloudTrailLogs`, must be deleted manually, or renamed, before you set up another landing zone.
- The two S3 buckets with reserved names for logs must be removed, or renamed, manually.
- You must delete, or rename, the existing Core and Custom organizations manually.

   **Note**
   Before you can delete the AWS Control Tower Core OU organization, you must first delete the logging and audit accounts, but not the master account. To delete these accounts, you must Sign in as a Root User (p. 23) to the audit account and to the logging account and delete them individually.

- You may wish to delete the AWS Single Sign-On (AWS SSO) configuration for AWS Control Tower manually, but you can proceed with the existing AWS SSO configuration.
- You may wish to remove the VPC created by AWS Control Tower, and remove the associated AWS CloudFormation stack set.

Setup After Decommissioning a Landing Zone

After you decommission your landing zone, you cannot successfully execute setup again until manual cleanup is complete. Also, without manual cleanup of these remaining resources, you may incur unexpected billing charges. You must attend to these issues:

- The AWS Control Tower master account is part of the AWS Control Tower **Root OU**. Be sure that these IAM roles and IAM policies are removed from the master account:
  - Roles:
    - AWSControlTowerAdmin
    - AWSControlTowerCloudTrailRole
    - AWSControlTowerStackSetRole
  - Policies:
    - AWSControlTowerAdminPolicy
    - AWSControlTowerCloudTrailRolePolicy
    - AWSControlTowerStackSetRolePolicy
  - You may wish to delete or update the existing AWS SSO configuration for AWS Control Tower before you set up a landing zone again, but it is not required that you delete it.
• You may wish to remove the VPC created by AWS Control Tower.
• Setup fails if the email addresses specified for the logging or audit accounts are associated with an existing AWS account. You must close the AWS accounts, or use different email addresses to set up a landing zone again.
• Setup fails if S3 buckets with the following reserved names already exist in the logging account:
  • `aws-controltower-logs-{accountId}-{region}` (used for the logging bucket).
  • `aws-controltower-s3-access-logs-{accountId}-{region}` (used for the logging access bucket).

You must either rename or remove these buckets, or use a different account for the logging account.
• Setup fails if the master account has the existing log group, `aws-controltower/CloudTrailLogs`, in CloudWatch Logs. You must either rename or remove the log group.

**Note**
You cannot set up a new landing zone in an organization with top-level OUs named either **Core** or **Custom**. You must rename or remove these OUs to set up a landing zone again.
Troubleshooting

If you encounter issues while using AWS Control Tower, you can use the following information to resolve them according to our best practices. If the issues you encounter are outside the scope of the following information, or if they persist after you've tried to resolve them, contact AWS Support.

Landing Zone Launch Failed

Common causes of landing zone launch failure:

- Lack of response to a confirmation email message.
- AWS CloudFormation StackSet failure.

**Confirmation email messages:** If your master account is less than an hour old, you may encounter issues when the additional accounts are created.

**Action to take**

If you encounter this issue, check your email. You might have been sent confirmation email that is awaiting response. Alternatively, we recommend that you wait an hour, and then try again. If the issue persists, contact AWS Support.

**Failed StackSets:** Another possible cause of landing zone launch failure is AWS CloudFormation StackSet failure. AWS Security Token Service (STS) regions must be enabled in the master account for all AWS Regions in which AWS Control Tower is supported, so that the provisioning can be successful; otherwise, stack sets will fail to launch.

**Action to take**

Be sure to enable all of your required AWS Security Token Service (STS) endpoint regions before you launch AWS Control Tower.

Currently, AWS Control Tower is supported in the following AWS Regions:

- US East (N. Virginia)
- US East (Ohio)
- US West (Oregon)
- Europe (Ireland)
- Asia Pacific (Sydney)

New Account Provisioning Failed

If you encounter this issue, check for these common causes.

**When you filled out the account provisioning form, you may have:**

- specified tagOptions,
• enabled SNS notifications,
• enabled provisioned product notifications.

Try again to provision your account, without specifying any of those options. For more information, see Provisioning Account Factory Accounts With AWS Service Catalog (p. 65).

Other common causes for failure:
• If you created a provisioned product plan (to view resource changes), your account provisioning may remain in an In progress state indefinitely.
• Creation of a new account in Account Factory will fail while other AWS Control Tower configuration changes are in progress. For example, while a process is running to add a guardrail to an OU, Account Factory will display an error message if you try to provision an account.

To check the status of a previous action in AWS Control Tower
• Navigate to AWS CloudFormation > AWS StackSets
• Check each stack set related to AWS Control Tower (prefix: "AWSControlTower")
• Look for AWS StackSets operations that are still running.

If your account provisioning takes longer than one hour, it's best to terminate the provisioning process and try again.

Failed to Enroll an Existing Account

If you try once to enroll an existing AWS account and that enrollment fails, when you try a second time, the error message may tell you that the stack set exists. To continue, you must remove the provisioned product in Account Factory.

If the reason for the first enrollment failure was that you forgot to create the AWSControlTowerExecution role in the account in advance, the error message you'll receive correctly tells you to create the role. However, when you try to create the role, you are likely to receive another error message stating that AWS Control Tower could not create the role. This error occurs because the process has been partially completed.

In this case, you must take two recovery steps before you can proceed with enrolling your existing account. First, you must terminate the provisioned product in Account Factory. Next, you must use the AWS Organizations console to manually move the account out of the OU and back to the root. After that is done, create the AWSControlTowerExecution role in the account, and then fill in the Enroll account form again.

Unable to Update an Account Factory Account

You may encounter an issue that prevents you from updating a provisioned account. You can see an error message similar to this one: AWS Control Tower could not baseline VPC in the managed account because of existing resource dependencies.

Common cause: AWS Control Tower always removes the AWS default VPC during initial provisioning. To have an AWS default VPC in an account, you must add it after account creation. AWS Control Tower has its own default VPC that replaces the AWS default VPC, unless you set up Account Factory the way the
walkthrough shows you—so that AWS Control Tower doesn’t provision a VPC at all. Then the account has no VPC. You’d have to re-add the AWS default VPC if you want to use that one.

However, AWS Control Tower doesn’t support the AWS default VPC. Deploying one causes the account to enter a Tainted state. When it is in that state, you cannot update the account through AWS Service Catalog.

The Tainted state causes a follow-on issue: An account that is not updated may prevent enabling guardrails on the OU of which it is a part.

**Action to take:** You must delete the default VPC that you added, and then you will be able to update the account.

---

**Failure Error that Mentions AWS Config**

If AWS Config is enabled in any AWS Region supported by AWS Control Tower, you may receive an error message because a pre-check has failed. The message might not seem to explain the problem adequately, due to some underlying behavior of AWS Config.

**You may receive an error message, similar to one of these:**

- AWS Control Tower cannot create an AWS Config delivery channel because one already exists. To continue, delete the existing delivery channel and try again.

- AWS Control Tower cannot create an AWS Config configuration recorder because one already exists. To continue, delete the existing delivery channel and try again.

**Common cause:** When the AWS Config service is enabled on an AWS account, it creates a configuration recorder and delivery channel with a default naming. If you disable the AWS Config service through the console, it does not delete the configuration recorder or the delivery channel. You must delete them through the CLI. If the AWS Config service is enabled in any one of the Regions supported by AWS Control Tower, it can result in this failure.

**Action to take:** Delete the configuration recorder and delivery channel in all supported regions. Disabling AWS Config is not enough, the configuration recorder and delivery channel must be deleted by means of the CLI. After you’ve deleted the configuration recorder and delivery channel from the CLI, you can try again to launch AWS Control Tower and enroll the account.

Here are some example AWS Config CLI commands you can use to determine the status of your configuration recorder and delivery channel.

**View commands:**

- `aws configservice describe-delivery-channels`
- `aws configservice describe-delivery-channel-status`
- `aws configservice describe-configuration-recorders`
- The normal response is something like "name": "default"

**Delete commands:**

- `aws configservice stop-configuration-recorder --configuration-recorder-name NAME-FROM-DESCRIPTION-OUTPUT`
No Launch Paths Found Error

When you're trying to create a new account, you may see an error message similar to this one:

No launch paths found for resource: prod-dpqfyywxxxx

This error message is generated by AWS Service Catalog, which is the integrated service that helps provision accounts in AWS Control Tower.

Common Causes:

- You may be logged in as root. AWS Control Tower does not support creating accounts when you're logged in as root.
- Your SSO user has not been added to the appropriate permission group.
- If you are authenticated as an IAM user, you must add it to the AWS Service Catalog portfolio so that it has the correct permissions.

Received an Insufficient Permissions Error

It's possible that your account may not have the necessary permissions to perform certain work in certain AWS Organizations. If you encounter the following type of error, check all the permissions areas, such as IAM or SSO permissions, to make sure your permission is not being denied from those places:

"You have insufficient permissions to perform AWS Organizations API actions."

If you believe your work requires the action you're attempting, and you can't locate any relevant restriction, contact your system administrator or AWS Support.

Detective guardrails are not taking effect on accounts

If you've recently expanded your AWS Control Tower deployment into a new AWS Region, newly-applied detective guardrails do not take effect on new accounts you create in any region until the individual accounts within OUs governed by AWS Control Tower are updated. Existing detective guardrails on existing accounts are still in effect.

For example, if you've recently updated your AWS Control Tower landing zone for release 2.3, which enables the Asia Pacific (Sydney) region, deploying the 2.3 release updates the landing zone, but it does not update each individual account.
If you have not updated your landing zone for AWS Control Tower release 2.3, and if you do not require your workloads to run in the Asia Pacific (Sydney) region, consider remaining at release 2.2. As long as release 2.2 is in effect, the behavior of deploying detective guardrails to accounts is unchanged.

**Action to take: Update accounts.**

To update multiple individual accounts, you can use the APIs from AWS Service Catalog and the AWS CLI to automate the updates. For more information about how to approach the update process, see this Video Walkthrough (p. 112). You can substitute the `UpdateProvisionedProduct` API for the `ProvisionProduct` API shown in the video.

If you have further difficulties with enabling detective guardrails on your accounts, contact AWS Support.

**AWS Support**

If you want to move your existing member accounts into a different support plan, you can sign in to each account with root account credentials, compare plans, and set the support level that you prefer.

We recommend that you update the MFA and account security contacts when you make changes to your support plan.
The following table describes important changes to the *AWS Control Tower User Guide*. For notifications about documentation updates, you can subscribe to the RSS feed.

<table>
<thead>
<tr>
<th>Update History Change</th>
<th>Update History Description</th>
<th>Update History Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>AWS Control Tower allows existing organizations to set up a landing zone (p. 127)</td>
<td>You can now launch a landing zone for AWS Control Tower in an existing organization, to bring the organization into governance. The <strong>Quick account provisioning</strong> capability in AWS Control Tower was renamed to <strong>Enroll account</strong> and it now permits enrollment of existing AWS accounts as well as creation of new accounts.</td>
<td>April 16, 2020</td>
</tr>
<tr>
<td>AWS Control Tower is now available in Asia Pacific (p. 127)</td>
<td>AWS Control Tower is now available to be deployed in the Asia Pacific (Sydney) AWS Region. This release requires manual updates to vended accounts, update only if you plan to run workloads in Asia Pacific (Sydney).</td>
<td>March 3, 2020</td>
</tr>
<tr>
<td>Decommissioning an AWS Control Tower landing zone is possible (p. 127)</td>
<td>AWS Support can help you permanently decommission a landing zone through a mostly automated process that preserves your organizations, although some manual cleanup is required.</td>
<td>February 27, 2020</td>
</tr>
<tr>
<td>Quick account provisioning is available in AWS Control Tower (p. 127)</td>
<td>Quick account provisioning makes it easier to launch new member accounts when your landing zone is up to date, with the <strong>Enroll account</strong> feature.</td>
<td>February 20, 2020</td>
</tr>
<tr>
<td>Lifecycle events are tracked in AWS Control Tower (p. 127)</td>
<td>Lifecycle events provide additional details for certain AWS Control Tower events, to make some workflow automation easier.</td>
<td>December 12, 2019</td>
</tr>
<tr>
<td>Settings and Activities pages are available for AWS Control Tower (p. 127)</td>
<td>The Settings and Activities pages make it easier to update your landing zone and to view logged events.</td>
<td>November 30, 2019</td>
</tr>
<tr>
<td>Additional preventive guardrails are available for AWS Control Tower (p. 127)</td>
<td>Preventive guardrails in AWS Control Tower keep your organization and resources aligned with your environment.</td>
<td>September 6, 2019</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>Additional detective guardrails are available for AWS Control Tower (p. 127)</td>
<td>Detective guardrails in AWS Control Tower give information about the state of your organization and resources.</td>
<td>August 27, 2019</td>
</tr>
<tr>
<td>AWS Control Tower is now generally available (p. 127)</td>
<td>AWS Control Tower is a service that offers the easiest way to set up and govern your multi-account AWS environment at scale.</td>
<td>June 24, 2019</td>
</tr>
</tbody>
</table>
AWS glossary

For the latest AWS terminology, see the AWS glossary in the AWS General Reference.