

Implementation Guide

Network Orchestration for AWS Transit Gateway



Network Orchestration for AWS Transit Gateway: Implementation Guide

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon's trademarks and trade dress may not be used in connection with any product or service that is not Amazon's, in any manner that is likely to cause confusion among customers, or in any manner that disparages or discredits Amazon. All other trademarks not owned by Amazon are the property of their respective owners, who may or may not be affiliated with, connected to, or sponsored by Amazon.

Table of Contents

Solution overview	1
Features and benefits	2
Use cases	3
Concepts and definitions	3
Architecture overview	5
Architecture diagram	5
AWS Well-Architected design considerations	8
Operational excellence	8
Security	8
Reliability	9
Performance efficiency	9
Cost optimization	9
Sustainability	10
Architecture details	11
Automated approval	11
Manual approval	12
Transit Gateway inter-Region peering	14
AWS services in this solution	15
Plan your deployment	17
Cost	17
Sample cost table	17
Security	19
IAM roles	20
AWS WAF	20
Amazon CloudFront	20
Amazon Cognito	20
Supported AWS Regions	21
AWS accounts for multi-account environments	22
Quotas	22
Quotas for AWS services in this solution	22
CloudFormation quotas	23
Lambda quotas	23
Transit Gateway quotas	23
AWS Transit Gateway Network Manager quotas	23

Deploy the solution	24
Prerequisites	24
Activate AWS RAM for Organizations accounts	24
Identify the Organizations ARN	25
Deployment process overview	25
AWS CloudFormation templates	27
Step 1: Launch the organization role stack (optional)	28
Step 2: (Optional) Launch the service-linked role for AWS RAM hub stack	30
Step 3: Launch the hub stack	31
Step 4: Launch the spoke stack(s)	42
Step 5: Add tags	44
Transit gateway attachments to a VPC	44
Transit gateway peering attachments	48
Custom compliance for network changes	50
Update the solution	56
Update the organization role stack (optional)	56
Update the hub stack(s)	57
Hub stack	57
Service-linked role for AWS RAM hub stack	58
Update the spoke stack(s)	58
Spoke stack	58
Upgrade from nested stack to consolidated spoke stack.	60
Migrate the hub stack (optional)	60
Migrate from v1 to v3	60
Migrate from v2 to v3	61
Troubleshooting	63
Problem: Amazon EventBridge event bus permissions error	63
Resolution	64
Problem: AlreadyExists error	64
Resolution	64
Contact AWS Support	64
Create case	64
How can we help?	64
Additional information	65
Help us resolve your case faster	65
Solve now or contact us	65

Uninstall the solution	66
Using the AWS Management Console	66
Using AWS Command Line Interface	66
Manually delete resources	66
Use the solution	67
Use the web UI	67
Sign in to the web UI	67
Manage network activities	68
Using and customizing route tables	69
Default route tables	69
Custom route tables	71
On-premises connectivity	74
Developer guide	75
Source code	75
Reference	76
Data collection	76
Contributors	76
Revisions	77
Notices	78

Automate setting up and managing your transit networks with AWS Transit Gateway

The Network Orchestration for AWS Transit Gateway solution automates the process of setting up and managing transit networks in multi-account AWS environments. The solution creates a web user interface (UI) to help you control, audit, and approve or reject transit network changes. This solution supports both [AWS Organizations](#) and standalone AWS accounts, and you can use the solution to visualize your transit network across multiple AWS Regions. You can use this solution with the default deployment template or customize it to meet your specific use case.

You can use [AWS Transit Gateway](#) to attach [Amazon Virtual Private Clouds](#) (Amazon VPCs) in the same AWS Region, and to route traffic between them. With this solution, you can connect your VPCs across multiple accounts by tagging the VPCs. It also connects your transit gateway across multiple AWS Regions by tagging the transit gateway. You can set rules to automatically approve or reject, or manually approve, the network changes.

This implementation guide provides an overview of the Network Orchestration for AWS Transit Gateway solution, its reference architecture and components, considerations for planning the deployment, and configuration steps for deploying the solution to the Amazon Web Services (AWS) Cloud.

The intended audience for implementing this solution in their environment includes solution architects, networking professionals, business decision makers and cloud professionals. To deploy and use this solution, you should have an understanding of Amazon VPC, route tables, subnets, transit gateways, and network protocols. For additional training about these topics, see [AWS Networking Basics](#), [Understanding AWS Networking Gateways](#), and [Advanced Architecting on AWS](#).

Use this navigation table to quickly find answers to these questions:

If you want to ...	Read ...
Know the cost for running this solution.	Cost
The estimated cost for running this solution in the US East (N. Virginia) Region is USD \$85.22 per month.	

If you want to ...	Read ...
Understand the security considerations for this solution.	Security
Know how to plan for quotas for this solution.	Quotas
Know the supported AWS Regions for this solution.	Supported AWS Regions
View or download the AWS CloudFormation template included in this solution to automatically deploy the infrastructure resources (the "stack") for this solution.	AWS CloudFormation template
Access the source code and optionally use the AWS Cloud Development Kit (AWS CDK) to deploy the solution.	GitHub repository

Features and benefits

This solution provides the following features:

Cross-account and cross-Region integration

This solution helps you automate the management of your networks across multiple AWS accounts and AWS Regions (through inter-Region peering). This helps reduce the time that you need to configure connectivity through your AWS environment.

Change management

For critical and sensitive environments, you can enable manual approval workflows through the web UI to accept or reject connectivity requests between your environments.

Tracking and auditing

Use the web UI to track or audit changes to your network environment, and to review approved or rejected requests.

Compliance

Use rules to automatically accept or reject network changes based on the organizational unit. For more information about approvals, see the [Automated approval](#) and [Manual approval](#) workflows.

Use cases

Network connectivity

To meet your workloads' requirements, this solution helps you attach VPCs with Transit Gateway by tagging the VPCs and subnets across multiple accounts. Based on the VPC and subnet tags, the solution automatically updates the subnet's associated route table with default routes to the transit gateway. It also creates association and enables propagation in the transit gateway route tables.

To connect your network across AWS Regions, this solution can create inter-Region transit gateway peering attachments.

Concepts and definitions

This section describes key concepts and defines terminology specific to this solution.

application

A logical group of AWS resources that you want to operate as a unit.

attachment

Connection from a resource to a transit gateway. For this solution, you can attach one or more VPCs to the transit gateway.

CloudFormation stack

Provisions the resources that are described in the templates.

CloudFormation template

Specifies the AWS resources included in this solution and their properties.

hub account

Central account where the solution is deployed and manages your central transit gateway. This is typically your network account.

network account

The networking account serves as the central hub for your network on AWS. You can manage your networking resources and route traffic between accounts in your environment, your on-premises, and egress/ingress traffic to the internet.

route table

A set of routing rules that controls the traffic leaving any subnet that's associated with the route table. This includes dynamic and static routes that decide the next hop based on the destination IP address of the packet.

state machine

A workflow for [AWS Step Functions](#).

Note

For a general reference of AWS terms, see the [AWS Glossary](#).

Architecture overview

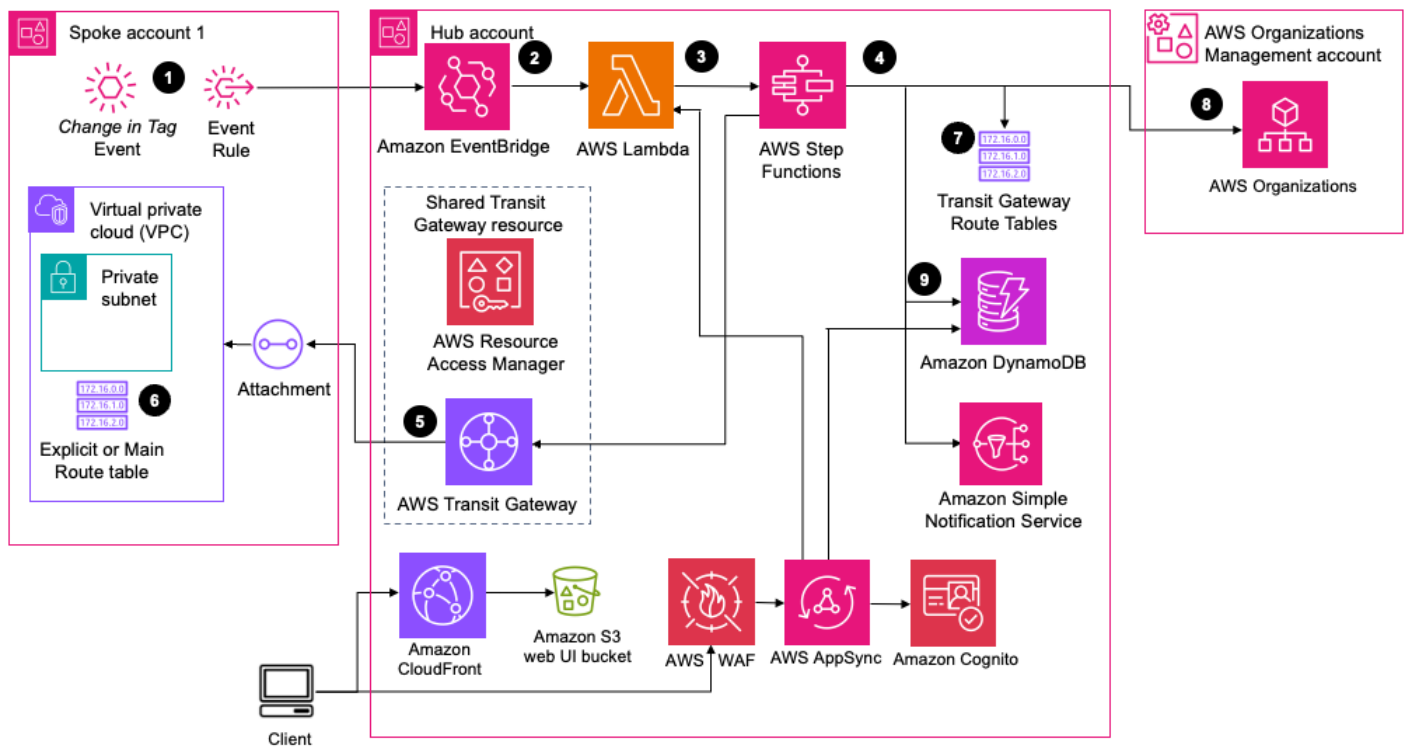
This section provides a reference implementation architecture diagram for the components deployed with this solution. This solution includes:

- A CloudFormation hub template (`aws-transit-network-orchestrator-hub.template`) that you deploy in the [hub account](#). This template launches all the components necessary to automatically connect your VPCs to Transit Gateway. The template also deploys a web UI. For recommendations on choosing a hub account, refer to [AWS accounts](#).
- A CloudFormation spoke template (`aws-transit-network-orchestrator-spoke.template`) to deploy in your spoke account(s).
- A CloudFormation organization role template (`aws-transit-network-orchestrator-organization-role.template`) to optionally deploy in your Organizations management account.

Architecture diagram

Deploying this solution with the default parameters deploys the following components in your AWS account.

Architecture diagram of AWS resources deployed to automate managing Transit Gateway attachments.



Note

CloudFormation resources are created from AWS Cloud Development Kit (AWS CDK) constructs.

1. This template deploys an [Amazon EventBridge](#) rule that monitors specific VPC and subnet tag changes.

Note

To identify the VPCs (spoke accounts) for the solution to manage, [tag the VPCs](#) and the selected subnets within those VPCs.

2. An EventBridge rule in the spoke account sends the tags to the EventBridge bus in the hub account.
3. The rules associated with the EventBridge bus invoke an [AWS Lambda function](#) to start the solution workflow. For more information about workflows, refer to [Architecture details](#).

Note

Wait for the hub stack launch to complete before you launch spoke templates. The spoke accounts depend on the EventBridge bus that's created during the hub stack launch.

4. [AWS Step Functions](#) (solution [state machine](#)) processes network requests from the spoke accounts.
5. The state machine workflow attaches a VPC to the transit gateway.
6. The state machine workflow updates the VPC route table associated with the tagged subnet.
7. The state machine workflow updates the transit gateway route table with association and propagation changes.

Note

This workflow only updates the transit gateway route table defined in the VPC tags.

8. (Optional) The state machine workflow updates the attachment name with the VPC name and the Organizational Unit (OU) name for the spoke account (retrieved from the Org Management account).

Note

This occurs only if you provide your Organizations ARN for the **Account List or AWS Organizations ARN** template parameter. For more information, see [Step 3: Launch the hub stack](#).

9. The solution updates [Amazon DynamoDB](#) with the information extracted from the event and resources created, updated, or deleted in the workflow.

Users can view tagging event details and the history of network requests from different accounts, and monitor their status in the web UI. Administrators can accept or reject requests when manual approval is required.

AWS Well-Architected design considerations

This solution uses the best practices from the [AWS Well-Architected Framework](#) which helps customers design and operate reliable, secure, efficient, and cost-effective workloads in the cloud.

This section describes how the design principles and best practices of the Well-Architected Framework benefit this solution.

Operational excellence

This section describes how the principles and best practices of the [operational excellence pillar](#) benefit this solution.

- The solution pushes metrics to CloudWatch to provide observability into the infrastructure, including Lambda functions, Step Functions, [Amazon API Gateway](#), [Amazon Simple Storage Service](#) (Amazon S3) buckets, and the rest of the solution components.
- AWS X-Ray traces are enabled for Step Functions and AWS AppSync. This helps you visualize the components of the state machine and analyze user requests as they travel through your AWS AppSync APIs to the underlying services, identify performance bottlenecks, and troubleshoot requests that resulted in an error.

Security

This section describes how the principles and best practices of the [security pillar](#) benefit this solution.

- The web UI users are authenticated and authorized with [Amazon Cognito](#).
- All inter-service communications use [AWS Identity and Access Management](#) (IAM) roles.
- All multi-account communications use IAM roles.
- All IAM roles used by the solution follow least-privilege access. In other words, they only contain minimum permissions required so that the service can function properly.
- The access token obtained from Amazon Cognito is used to authorize application programming interface (API) calls.
- All data storage, such as S3 buckets and DynamoDB tables, have encryption at rest.
- AWS WAF protects the web UI and APIs from attacks using solution-configured web access control lists (ACLs).

The solution creates CloudFront distribution with the [Default CloudFront SSL Certificate](#) which allows TLS 1.1 and TLS 1.0. We recommend using Custom SSL Certificate with [TLSv1.2_2021 security policy](#) to disallow insecure protocols and cipher suites.

Reliability

This section describes how the principles and best practices of the [reliability pillar](#) benefit this solution.

- The solution uses serverless AWS services wherever possible (such as Lambda, [AWS AppSync](#), Amazon S3, and Step Functions) to ensure high availability and recovery from service failure.
- AWS protects the solution against definition errors of state machines leveraged by Step Functions by running automated tests on the solution.
- Data processing uses Lambda functions. The solution stores data in DynamoDB and Amazon S3, so it persists in multiple Availability Zones by default.

Performance efficiency

This section describes how the principles and best practices of the [performance efficiency pillar](#) benefit this solution.

- The solution uses serverless architecture. For additional details, refer to [Reliability](#).
- The solution uses `error handling` in Step Functions to run concurrent state machine executions that add or remove multiple subnets in the VPC-TGW attachment. This allows you to create VPC and related resources in parallel using CloudFormation stack.
- You can launch the solution in any AWS Region that supports the AWS services used in this solution (such as Lambda, API Gateway, Amazon S3, Step Functions, Amazon Cognito, [Amazon CloudFront](#), and [AWS WAF](#)). You can also choose not to deploy the web UI if CloudFront and Amazon Cognito aren't supported in the Region. Refer to [Supported AWS Regions](#).
- AWS automatically tests and deploys the solution daily. Our solution architects and subject matter experts review the solution for areas to experiment and improve.

Cost optimization

This section describes how the principles and best practices of the [cost optimization pillar](#) benefit this solution.

- The solution uses serverless architecture (for example, Step Functions and DynamoDB) to minimize the cost of unused compute infrastructure, and customers pay only for what they use.
- The compute layer defaults to Lambda, which uses a pay-per-use model.

Sustainability

This section describes how the principles and best practices of the [sustainability pillar](#) benefit this solution.

- The solution uses managed and serverless services to minimize the environmental impact of the backend services.
- The solution's serverless design is aimed at reducing carbon footprint compared to the footprint of continually operating on-premises servers.
- The web UI allows users to select scan parameters to perform selective scans in specific AWS accounts, Regions, and services.

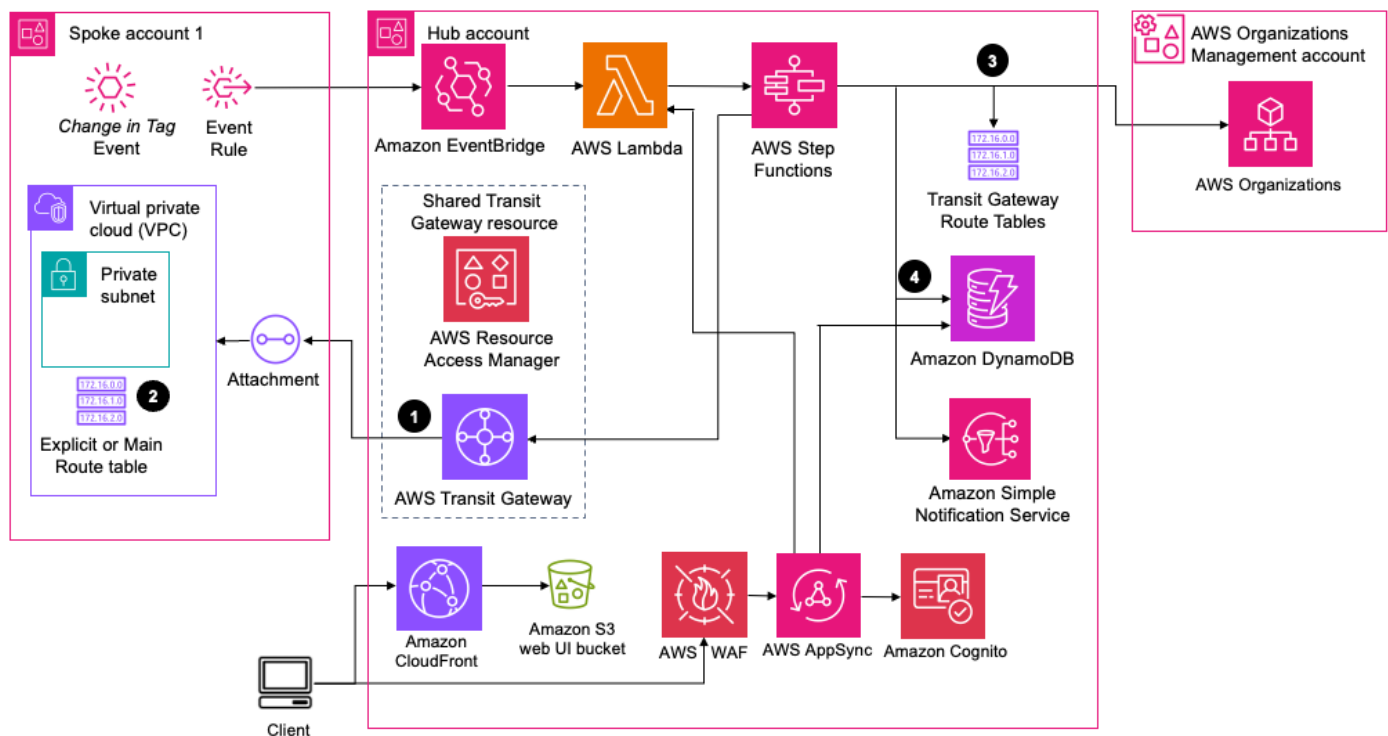
Architecture details

This section describes the components and AWS services that make up this solution and the architecture details on how these components work together.

Automated approval

By default, the solution approves network requests from spoke accounts automatically. This section provides detail about this workflow.

Architecture diagram of AWS resources deployed to approve network requests automatically.



- Depending on the event, the state machine can perform the following actions:
 - Create, update, or delete transit gateway attachments to the VPC
 - Create or update transit gateway route table associations
 - Enable or disable transit gateway route table propagations
- The state machine creates routes in the VPC route tables associated with the subnets that you tagged, with the following exceptions (see [Step 5. Add tags](#) for more information):

- If there is no explicit route table associated with the subnet, the solution updates the main route table instead.
 - If you tag a second subnet in the same Availability Zone, you must use the `route-to-tgw` tag key to only add the route and skip adding the subnet in the attachment.
3. The state machine then adds a new status tag to the VPC or the subnet with the status of the request.
 4. The state machine updates the DynamoDB table to activate the network administrator to audit the network change history. The changes in DynamoDB are automatically reflected in the web UI dashboard. Administrators and users can sign in to the web UI to review the history of all changes that occurred in the network.

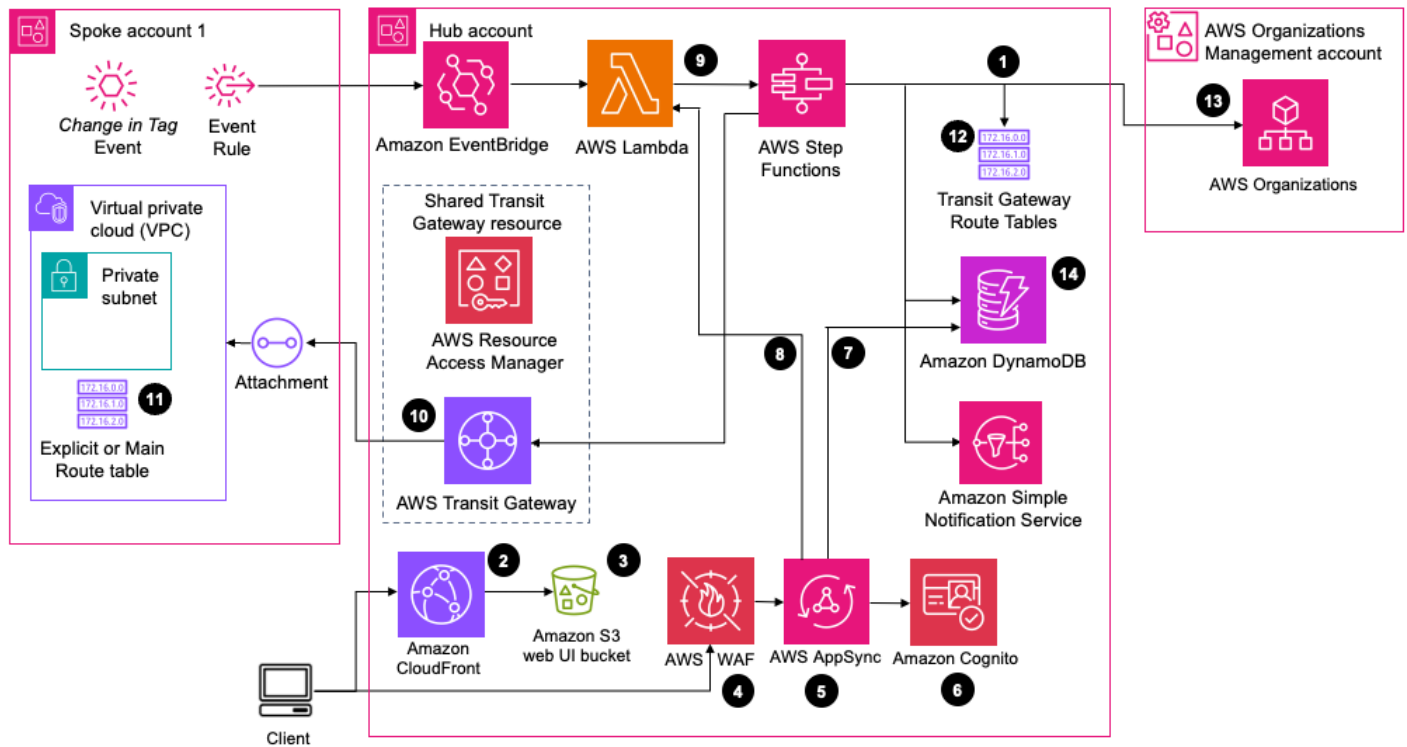
Manual approval

You can choose to manually approve network requests from spoke accounts, instead of automated approval. This section provides detail about this workflow.

Important

If you don't deploy the UI, you can't approve or reject a network change. All the network changes will be auto-approved. You can use the compliance rules to automatically approve and reject network changes.

Architecture diagram of AWS resources deployed to support manual approval of network requests.



1. If you set the **ApprovalRequired** tag key to Yes or Conditional in the **Transit gateway route table** parameter, the state machine skips changes depending on the rules set under the Conditional setting. To set up this flag, refer to [Transit Gateway route table tags](#).
2. The administrator signs in to the web UI, and the Amazon Cognito [user pool](#) authenticates each user. CloudFront delivers the web UI content from an S3 bucket.
3. The S3 bucket hosts the web UI.
4. The web UI gets a token from Amazon Cognito and sends a request to AWS AppSync. AWS WAF protects the APIs from security events. This solution configures a set of rules called a web ACL. The web ACL allows, blocks, or counts web requests based on configurable, user-defined web security rules and conditions.
5. AWS AppSync provides the solution's API layer using GraphQL.
6. Amazon Cognito authenticates the token in the header of the API requests.
7. An AWS AppSync [resolver](#) updates the DynamoDB table with the processing status.
8. An AWS AppSync resolver invokes a Lambda function that validates the event.
9. A Lambda function starts a new state machine execution.
10. The state machine workflow attaches a VPC to the transit gateway.
11. The state machine workflow updates the VPC route table associated with the tagged subnet.

12.The state machine workflow updates the transit gateway route table with association and propagation changes.

Note

This workflow only updates the transit gateway route table defined in the VPC tags.

13(Optional) The state machine workflow updates the attachment name with the VPC name and the Organizational Unit (OU) name for the spoke account (retrieved from the Org Management account).

Note

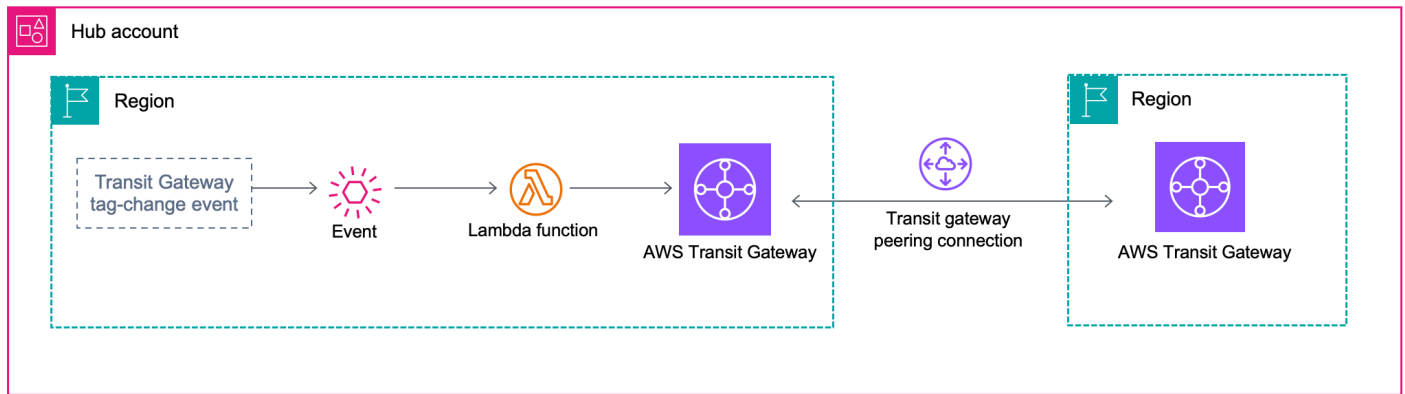
This occurs only if you provide your Organizations ARN for the **Account List or AWS Organizations ARN** template parameter. For more information, see [Step 3: Launch the hub stack](#).

14.The solution updates the DynamoDB with the information extracted from the event and resources created, updated, or deleted in the workflow. The changes in DynamoDB are automatically reflected in the web UI dashboard. Administrators and users can sign in to the web UI to review the history of all changes that occurred in the network.

Transit Gateway inter-Region peering

You can use Transit Gateway peering to directly route traffic between two transit gateways in the same AWS Region or across Regions. This section provides information about how this solution supports peering.

Architecture diagram of AWS resources deployed to support Transit Gateway inter-Region peering.



1. When you [tag the transit gateway](#), an EventBridge event initiates. The target for this event is the transit gateway peering attachment Lambda function in the hub account.
2. The tgw-peering Lambda function creates the peering attachment between the transit gateways based on the tag key and value. The peering attachment state transitions from `InitializingRequest` to `PendingAcceptance`.
3. The Lambda function accepts the peering attachment request in the remote Region.
4. The solution sets the peering attachment state to `Available`.

AWS services in this solution

AWS service	Description
AWS Transit Gateway	Core. Deploys a transit gateway that connects VPCs through a central hub.
AWS Lambda	Core. Deploys multiple Lambda functions to support core microservices and create transit gateway attachments.
AWS Step Functions	Core. Deploys a state machine to orchestrate the subnet and VPC tagging events and create transit gateway attachments.

AWS service	Description
Amazon DynamoDB	Core. Deploys a DynamoDB table for VPC and transit gateway attachments, and for transit gateway peering attachments.
Amazon EventBridge	Core. Deploys an event bus and event rules to connect components of the solution.
AWS X-Ray	Supporting. Deploys traces for API Gateway and Step Functions, allowing you to investigate root causes of failures.
Amazon SNS	Optional. Deploys a topic that sends an email notification with the optional web UI URL.
Amazon Cognito	Optional. Deploys a user pool that supports identity authentication for the optional web UI.
AWS AppSync	Optional. Deploys AWS AppSync schema and resolvers for the DynamoDB table and Lambda functions. Using resolvers, AWS AppSync translates GraphQL requests and fetches information from DynamoDB.
Amazon S3	Optional. Deploys Amazon S3 buckets to host the web UI assets.
AWS WAF	Optional. Deploys AWS WAF web access control list (ACL) to protect AWS AppSync from common security events, such as SQL injection and cross-site scripting (XSS).
Amazon CloudFront	Optional. Deploys CloudFront with an Amazon S3 bucket as the origin. This restricts access to the Amazon S3 bucket so that it's not publicly accessible and prevents direct access from the bucket.

Plan your deployment

This section describes the [cost](#), [architecture](#), [network security](#), and other considerations before deploying the solution.

Cost

You are responsible for the cost of the AWS services used while running this solution. As of this revision, the cost for running this solution with the default settings in the US East (N. Virginia) Region is approximately **\$85.22 a month**. These costs are for the resources shown in the [Sample cost table](#).

See the pricing webpage for each AWS service used in this solution.

We recommend creating a [budget](#) through [AWS Cost Explorer](#) to help manage costs. Prices are subject to change. For full details, see the pricing webpage for each AWS service used in this solution.

Sample cost table

The following table provides a sample cost breakdown for deploying this solution with the default parameters in the US East (N. Virginia) Region for one month. This cost estimate assumes the following:

- The solution manages two VPCs attached to one transit gateway, with each VPC containing two subnets in different Availability Zones.
- The solution makes automated queries to DynamoDB from an actively running web UI every five minutes. This estimate does not include manual queries.
- One GB of data a month travels between the two VPCs through the transit gateway.
- The number of requests to the GraphQL API is 10,000 a month.

AWS service	Dimensions	Cost [USD]
Variable Costs		
Transit Gateway	Hourly charge (containing two VPC attachments)	\$72.00

AWS service	Dimensions	Cost [USD]
Transit Gateway	Data processing charge (data transfer of 1 GB from two attached VPCs)	\$0.60
Transit Gateway	Data processing and outbound inter-Region transfer charge (data transfer of 1 GB between two inter-Region peered transit gateways)	\$0.40
Amazon DynamoDB	Includes automated queries only	\$3.27
AWS AppSync	Includes auto approval workflow only	\$1.23
Fixed Costs		
Amazon EventBridge		< \$ 0.01
AWS WAF		\$ 7.61
AWS Step Functions	State transitions: 100-120 transitions per month (5-6 workflow executions × 20 transitions each). State machine has 20 Task states that invoke Lambda	< \$ 0.01

AWS service	Dimensions	Cost [USD]
AWS Lambda	Duration: 700-850 GB-seconds per month (StateMachineLambdaFunction: 100-120 invocations at 4 sec with 1.5 GB, CustomResourceLambda: 2-4 invocations at 10 sec with 1.5 GB, MetricsCollectorLambda: 30 invocations at 5 sec with 0.5 GB)	< \$ 0.01
AWS X-Ray	100,000 Traces recorded for 2 services (Step Functions and AppSync) with default 5% sampling rate	< \$ 0.10
Total:		~ \$85.22 / month

Note

AWS Step Functions state transitions and AWS Lambda duration charges are included in this estimate. With the assumed usage pattern (5-6 workflow executions per month for 2 VPCs with minimal tag changes), both services operate within AWS Free Tier limits (4,000 state transitions/month and 400,000 GB-seconds/month), resulting in negligible charges. For environments with more frequent network changes, costs may increase proportionally but remain minimal.

Security

When you build systems on AWS infrastructure, security responsibilities are shared between you and AWS. This [shared responsibility model](#) reduces your operational burden because AWS operates, manages, and controls the components including the host operating system, the virtualization

layer, and the physical security of the facilities in which the services operate. For more information about AWS security, visit [AWS Cloud Security](#).

IAM roles

IAM roles allow customers to assign granular access policies and permissions to services and users on AWS. This solution creates IAM roles and sets permissions in the respective accounts. This allows the solution to assume a defined role in the spoke and management account to make changes when necessary. The hub account assumes role in the Management account and spoke accounts.

AWS WAF

AWS WAF is a web application firewall that helps protect web applications and APIs from attacks. It allows you to configure a web ACL that allows, blocks, or counts web requests based on configurable web security rules and conditions that you define. For more information, refer to [How AWS WAF Works](#).

You can use AWS WAF to protect AWS AppSync from common security events, such as SQL injection and XSS. These types of security events could affect API availability and performance, compromise security, or consume excessive resources. For example, you can create rules to allow or block requests from specified IP address ranges, requests from Classless Inter-Domain Routing (CIDR) blocks, requests that originate from a specific country or Region, requests that contain malicious SQL code, or requests that contain malicious script.

Amazon CloudFront

This solution deploys a static website [hosted](#) in an S3 bucket. To help reduce latency and improve security, this solution includes an Amazon CloudFront distribution with an origin access identity. This identity is a CloudFront user that helps provide public access to the solution's website bucket contents. For more information, refer to [Restricting access to an Amazon S3 origin](#).

Amazon Cognito

This solution creates Amazon Cognito user accounts for signing in to the web UI. The solution also grants the administrator and the read-only users with the appropriate permissions to control user access to data.

⚠ Important

If you connect an external identity provider through SAML, every user from your identity provider will have read access to the web UI. To prevent giving read access to all users by default, modify the `cognito-trigger` Lambda function deployed by this solution. For more information, see [Configuring Lambda function options](#).

Supported AWS Regions

This solution uses the AWS services that aren't currently available in all AWS Regions. You must launch this solution in an AWS Region where [these services](#) available. For the most current availability of AWS services by Region, see the [AWS Regional Services List](#).

This solution is available in the following AWS Regions:

Region name	
US East (Ohio)	Canada (Central)
US East (N. Virginia)	China (Beijing)
US West (Northern California)	China (Ningxia)
US West (Oregon)	Europe (Frankfurt)
Asia Pacific (Mumbai)	Europe (Ireland)
Asia Pacific (Seoul)	Europe (London)
Asia Pacific (Singapore)	Europe (Paris)
Asia Pacific (Sydney)	Europe (Stockholm)
Asia Pacific (Tokyo)	South America (Sao Paulo)

AWS accounts for multi-account environments

To deploy a multi-account environment, we recommend the following AWS account guidelines for each stack:

- **Hub stack** - Deploy to a member account in your AWS Organization, preferably where you have an existing transit gateway or plan to create a new one, or in a dedicated [network account](#) where you plan to create a new transit gateway. It can't be the Organizations management account.
- **Spoke stack** - Deploy to all the member accounts in your AWS Organization that have a VPC that you plan to attach to the transit gateway hub account. You must deploy it in the hub account if you want to attach VPCs in the hub account.
- **Organization role stack** - Optionally deploy in the Organizations management account to allow the solution to add Organizational Unit paths in the attachment name to help you identify the VPC location.

Quotas

Service quotas, also referred to as limits, are the maximum number of service resources or operations for your AWS account.

Quotas for AWS services in this solution

Make sure you have sufficient quota for each of the [services implemented in this solution](#). For more information, refer to [AWS service quotas](#).

Select one of the following links to go to the page for that service. To see the service quotas for all AWS services in the documentation without switching pages, view the information in the [Service endpoints and quotas](#) page in the *AWS General Reference guide* PDF instead.

Transit Gateway	AWS X-Ray
Lambda	Amazon SNS
Step Functions	Amazon Cognito
DynamoDB	AWS AppSync
EventBridge	Amazon S3

[AWS WAF](#)[CloudFront](#)

CloudFormation quotas

Your AWS account has [CloudFormation](#) quotas that you must be aware of when launching the stacks for this solution. By understanding these quotas, you can avoid limitation errors that can prevent you from deploying this solution successfully. For more information, refer to [AWS CloudFormation quotas](#) in the *AWS CloudFormation Users Guide*.

Lambda quotas

In the hub account, the state machine invokes Lambda functions to run the scan in parallel depending on the VPCs and subnets tagged across multiple accounts in your organization. [Review](#) and [increase](#) your Lambda invocation limit to avoid throttling.

Transit Gateway quotas

The solution creates a new transit gateway for each hub stack deployment unless you provide an existing transit gateway in the hub template parameter **(Optional) Do you wish to use an existing transit gateway? If yes, you must provide the transit gateway id below**. Your account has a default Transit Gateway quota of five.

AWS Transit Gateway Network Manager quotas

The solution creates a new [global network](#) for each hub stack deployment unless you provide an existing global network ID in the hub template parameter **(Optional) Do you wish to use an existing global network? If yes, you must provide the global network id below**. Your account has default global network quota of five. Only one global network is recommended for all the other deployments in different AWS Regions in the hub account. Provide the global network ID created by the first deployment in the other deployments in different AWS Regions.

Deploy the solution

This solution uses [CloudFormation templates and stacks](#) to automate its deployment. The CloudFormation templates specify the AWS resources included in this solution and their properties. The CloudFormation stack provisions the resources that are described in the templates.

Note

If you have previously deployed this solution, see [Update the solution](#) for update instructions.

Prerequisites

You must meet the following prerequisites before launching the stacks.

If your accounts are part of Organizations, you must first manually activate AWS RAM in the Organizations console and obtain the Organizations management account ID and organization ID before deploying the solution templates.

Activate AWS RAM for Organizations accounts

Use the following procedure to activate AWS RAM using the AWS Organizations console.

1. Sign in to the [AWS Organizations console](#).
2. In the navigation pane, select **Settings**.
3. Navigate to **AWS RAM**, and select **Enable access**.

Use the following procedure to activate the sharing option in the AWS RAM console.

1. Sign in to the [AWS RAM console](#).
2. In the navigation pane, select **Settings**.
3. Choose ***Enable sharing*with AWS Organizations**.
4. Choose **Save settings**.

Identify the Organizations ARN

To use this solution with accounts connected to AWS Organizations, you must specify the AWS Organizations ARN when you launch the hub template. The ARN value consists of the AWS Organizations management account ID and the organization ID. You can build the ARN string manually if you have access to the AWS Organizations management account ID and the organization ID, or you can use the AWS Command Line Interface (AWS CLI) to query the Organization ARN.

Note

If you don't have access to the management account ID and the Organization ID, contact your organization's management account administrator.

Use the following procedure to build the Organizations ARN manually after you have the Organizations management account ID and the organization ID.

1. Sign in to the [AWS Organizations console](#) from your organization's management account.
2. Select **AWS accounts** from the navigation menu.
3. Identify the management account and record the **Account ID**.
4. Select **Settings** from the navigation menu.
5. Record the entry for **Organization ID**.
6. Use the following sample to manually build the Organization ARN. Replace the placeholders with your management account and organization IDs.

```
arn:<AWS_PARTITION>:organizations::<ORG_MANAGEMENT_ACCOUNT_ID>:organization/<ORG-ID>
```

To use the AWS CLI to query the ARN, use the [describe-organization](#) API call. To set up AWS CLI, refer to [Configuring the AWS CLI](#) in the *AWS Command Line Interface User Guide*.

Deployment process overview

Follow the step-by-step instructions in this section to configure and deploy the solution into your account.

Important

This solution includes data collection. We use this data to better understand how customers use this solution and related services and products. AWS owns the data gathered through this survey. Data collection is subject to the [AWS Privacy Notice](#).

Before you launch the solution, review the [cost](#), [architecture](#), [security](#), and other considerations discussed earlier in this guide. Follow the step-by-step instructions in this section to configure and deploy the solution into your account.

Time to deploy: Approximately 25 minutes

[Step 1. \(Optional\) Launch the organization role stack](#)

- Launch the CloudFormation template in your Organizations management account.
- Enter values for the required **HubAccount** parameter.

[Step 2. \(Optional\) Launch the service- linked role for AWS RAM hub stack](#)

Note

If the `AWSServiceRoleForResourceAccessManager` role already exists, skip this step.

- Launch the CloudFormation template in your hub account.

[Step 3. Launch the hub stack](#)

- Launch the CloudFormation template in your hub account.
- Enter values for the required **Account List or AWS Organizations ARN** parameter.
- If deploying the web UI, enter values for the following parameters: **Allowed Listed Ranges**, **Console Login Information Email**, and **Cognito Domain Prefix**.
- Review the other template parameters and adjust, if necessary.

[Step 4. Launch the spoke stack\(s\)](#)

- Launch the CloudFormation template into your spoke account(s).
- Enter a value for the required **Network (Hub) Account** parameter.

[Step 5. Add tags](#)

- Add the required tags to the spoke VPCs and subnets.
- Validate and view transit gateway attachments.

AWS CloudFormation templates

This solution uses CloudFormation to automate its deployment in the AWS Cloud. It includes the following AWS CloudFormation templates, which you can download before deployment.

Note

AWS CloudFormation resources are created from AWS CDK constructs.

[View template](#)

network-orchestration-hub.template - Use this template to launch the solution and all associated components in your AWS network hub account. The default configuration deploys the following:

- One transit gateway
- Four transit gateway route tables
- One global network in Transit Gateway network manager
- Step Functions (to orchestrate VPC and transit gateway attachments)
- One [AWS Resource Access Manager](#) (AWS RAM) resource share
- One optional web UI with the following resources:
 - One DynamoDB table
 - EventBridge event bus and rules
 - IAM roles
- One optional web UI for network management with the following resources:
 - One Amazon SNS topic

- AWS AppSync API with WAF
- One Amazon Cognito user pool
- One CloudFront distribution with a CloudFront function
- Amazon S3 buckets

View template

network-orchestration-hub-service-linked-roles.template - Optionally use this template to launch the service-linked role for AWS RAM in your hub account. This stack is optional because it fails if the `AWSServiceRoleForResourceAccessManager` role already exists in the hub account.

View template

network-orchestration-spoke.template - Use this template to launch the solution and all associated components in your spoke account(s). The default configuration deploys EventBridge rules and IAM roles.

View template

network-orchestration-organization-role.template - Use this template to create an IAM role in the Organizations management account. The hub account requires this role to create easily-identifiable names for the transit gateway attachments, using a combination of OU path and VPC name.

Step 1: Launch the organization role stack (optional)

Follow the step-by-step instructions in this section to configure and deploy the organization role stack into your Organizations management account. This optional step helps you add a OU path and VPC name in the attachment tags for tracking and auditing.

1. Sign in to your AWS Organizations management account using the AWS Management Console and select the button to launch the `network-orchestration-organization-role.template` AWS CloudFormation template.

Launch solution

2. Launch this template in the same Region as you plan to launch the hub and spoke templates. The organization role template launches in the US East (N. Virginia) Region by default.
3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, assign a name to your solution stack. For information about naming character limitations, see [IAM and AWS STS quotas](#) in the *AWS Identity and Access Management User Guide*.
5. For **Parameters**, review the parameters for the template and modify them as necessary. This stack uses the following default values.

Parameter	Default	Description
HubAccount	<Requires input>	The account ID for the hub account.

6. Choose **Next**.
7. On the **Configure stack options** page, choose **Next**.
8. On the **Review and create** page, review and confirm the settings. Choose the box acknowledging that the template creates IAM resources.
9. Choose **Submit** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should see a status of **CREATE_COMPLETE** in approximately three to four minutes.

 Note

After the stack deploys, record the ARN for the role from the **Outputs** tab of the stack. You need this ARN as input for the **Account List or AWS Organizations ARN** parameter in the hub template.

Step 2: (Optional) Launch the service-linked role for AWS RAM hub stack

Follow the step-by-step instructions in this section to configure and deploy the optional service-linked role for AWS RAM hub stack into your hub account.

Important

This stack deploys the service-linked role for AWS RAM. AWS RAM uses the service-linked role named `AWSServiceRoleForResourceAccessManager` when you enable sharing with AWS Organizations. This role grants permissions to the AWS RAM service to view organization details, such as the list of member accounts and which organizational units each account is in.

This stack is optional because it fails if the role already exists in the hub account. You can validate if this role exists by signing in to the IAM console, selecting **Roles** from the navigation menu, and entering `AWSServiceRoleForResourceAccessManager` in the search box. If this role already exists, skip this step.

The stack deployment will fail with following details in the CloudFormation events if it already exists.

Error Code: `AlreadyExists`

Message: Service role name `AWSServiceRoleForResourceAccessManager` has been taken in this account.

1. Sign in to the AWS Management Console with your AWS network hub account and select the button to launch the `network-orchestration-hub-service-linked-roles.template` CloudFormation template.

Launch solution

2. Launch this template in the same Region as the hub template. The template launches in the US East (N. Virginia) Region by default.
3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.

4. On the **Specify stack details** page, assign a name to your solution stack. For information about naming character limitations, see [IAM and AWS STS quotas](#) in the *AWS Identity and Access Management User Guide*.
5. Choose **Next**.
6. On the **Configure stack options** page, choose **Next**.
7. On the **Review and create** page, review and confirm the settings. Choose the box acknowledging that the template creates IAM resources.
8. Choose **Submit** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should see a status of **CREATE_COMPLETE** in approximately three to four minutes.

Step 3: Launch the hub stack

Follow the step-by-step instructions in this section to configure and deploy the hub stack into your hub account.

1. Sign in to the AWS Management Console with your AWS network hub account and select the button to launch the `network-orchestration-hub.template` CloudFormation template. +{

A blue button with rounded corners and a white border, containing the text "Launch solution" in white.

2. The template launches in the US East (N. Virginia) Region by default. To launch this solution in a different AWS Region, use the Region selector in the console navigation bar. See [Supported AWS Regions](#) for more information on selecting a Region.
3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, assign a name to your solution stack. For information about naming character limitations, see [IAM and AWS STS quotas](#) in the *AWS Identity and Access Management User Guide*.

For **Parameters**, review the parameters for the template and modify them as necessary. This stack uses the following default values.

Parameter	Default	Description
Account Structure Settings		
Principal Type	AWS Organization ARN	Choose whether to use the default Organization ARN or a list of accounts. For guidance, refer to AWS accounts .
Account List or AWS Organizations ARN	<i><Requires input></i>	To use Organizations, enter the Organization ARN to share the transit gateway with the principals. For example: <pre>arn:<AWS_PARTITION>:organizations::<ORG_MANAGEMENT_ACCOUNT_ID>:organization/<ORG-ID></pre> For additional guidance to identify the ARN value, refer to Identify the Organizations ARN. To use an account list, enter a comma-separated list of AWS account numbers. For example, 123456789012 .

Parameter	Default	Description
Allow External Principals	Yes	Choose whether to enable or disable sharing the transit gateway with principals outside the organization. NOTE: You must set this parameter to Yes if you're using the List of Accounts value for the Principal Type parameter.
(Optional) IAM Role ARN of Management Account	<i><Optional input></i>	To tag attachments with the account name and OU path, provide the ARN for the role in the management account which can be assumed by the hub account. Leave this value blank if you're deploying this solution in your management account.
Web UI Settings		
Web User Interface	Yes	Option to deploy web UI to manage and audit the changes in the network. Select No to skip creation of the Console bucket, CloudFront, Amazon Cognito user pool, AWS WAF, and other support resources. NOTE: If you select No for this parameter, skip the remaining parameters in the Web UI Settings section.

Parameter	Default	Description
Allow Listed Ranges	0.0.0.0/1,128.0.0.0/1	Comma-separated list of CIDR ranges allowed to access GraphQL API. Default allows the entire internet.
Cognito Domain Prefix	<i><Requires input></i>	The prefix to the Cognito hosted domain name that will be associated with the user pool. Must be unique per AWS Region and must not contain reserved word 'cognito'.
Console Login Information Email	<i><Requires input></i>	The email address of the administrator user for the web UI. After launch, the solution sends an email to this address with a temporary password for the web UI.
Admin Username	adminuser	The username for network administrators with full read and write permissions to the web UI.
Read-Only Username	readonlyuser	The username for users with read-only permission to the web UI.

Parameter	Default	Description
Set MFA for Cognito to '`ON' or '`OPTIONAL'	OPTIONAL	ON - Amazon Cognito users will need to set up multi-factor authentication (MFA) on first login. OPTIONAL - Amazon Cognito users may opt to set up MFA.
SAML Provider Name	<i><Optional input></i>	If you want to connect an external identity provider, specify a name that appears on the UI.
SAML Provider Metadata URL	<i><Optional input></i>	If you want to connect an external identity provider, enter the URL to the metadata file of your SAML-based identity provider. The URL must begin with <code>https://</code>.
Transit Gateway Settings		

Parameter	Default	Description
(Optional) Do you wish to use an existing transit gateway? If yes, you must provide the transit gateway id below.	<i><Optional input></i>	The existing transit gateway ID in the current Region. For example, <code>tgw-a1b2c3d4e5</code> . If you don't provide a value, the solution creates a new transit gateway. If you do provide a value, the solution uses your existing transit gateway. You must ensure that: • The existing transit gateway has enabled the <code>AutoAcceptSharedAttachments</code> flag. The solution doesn't create additional transit gateway route tables for you. For more information, see Create a transit gateway. • The existing transit gateway isn't already registered with an existing global network ID provided in the parameter (Optional) Do you wish to use an existing global network? If yes, you must provide the global network id below. If you're updating the solution, provide a value.

Parameter	Default	Description
(Optional) Do you wish to register the transit gateway with a global network?	Yes	Choose whether to register the transit gateway with the global network. NOTE: You must set this parameter to No if either of the following is true: The transit gateway managed by the solution is already registered with an existing global network. The global network is not available in your selected AWS Region. Refer to Region availability in the <i>AWS Global Networks for Transit Gateways User Guide</i>.

Parameter	Default	Description
(Optional) Do you wish to use an existing global network? If yes, you must provide the global network id below.	<i><Optional input></i>	You can skip this section if you chose No for the previous parameter. The existing global network ID. To register the transit gateway ID (see previous parameter), provide an existing global network ID. For example, global-network-01231231231231231 . If you don't provide a value, the solution creates a new global network. If you do provide a value, ensure that the existing transit gateway provided in the (Optional) Do you wish to use an existing transit gateway? If yes, you must provide the transit gateway id below. parameter isn't already registered with the existing global network ID. NOTE: If you use this solution in more than one Region, we recommended using the global network created by the first solution deployment to register all the transit gateways deployed by the solution

Parameter	Default	Description
		in all the Regions with the same global network.
VPC Route Table Settings		
Choose the type of destination for target Transit Gateway	All-traffic (0/0)	Specify the default route setting for the route table associated with the tagged subnets. Choose from All-traffic (0/0) , RFC-1918 (10/8, 172.16/12, 192.168/16) , Custom-Destinations , or Configure-Manually . NOTE: If the route already exists, the solution will not overwrite it.
If selected 'Custom-Destinations', provide a comma separated list of CIDR Blocks.	<Optional Input>	Option to provide CIDR block(s). For example, 192.168.1.0/24, 192.168.2.0/24 . NOTE: Optional if providing prefix list ID(s).
If selected 'Custom-Destinations', provide a comma separated list of Customer-managed Prefix List IDs.	<Optional input>	Option to provide customer-managed prefix list ID(s). For example, p1-abcd1234, p1-efgh5678 . NOTE: Optional if providing CIDR block(s).
Tag Settings		

Parameter	Default	Description
Tag key for subnets - Adds subnet to VPC attachment and add routes to route table associated with the tagged subnet.	Attach-to-tgw	Specify a custom tag key name to initiate the transit gateway attachment workflow. NOTE: After initial deployment, don't change this solution's default parameter. If you change this parameter after deployment, you must manually update the tags on your VPCs.
Tag key for subnets - Only adds routes to route table associated with the tagged subnet.	Route-to-tgw	Specify a custom tag key name to skip the transit gateway attachment workflow and only update route table associated with the subnet being tagged.
Tag key for TGW Route Table Association with TGW Attachment	Associate-with	Specify a custom tag key name to initiate the transit gateway route table association with the transit gateway attachment workflow. NOTE: After initial deployment, don't change this solution's default parameter. If you change this parameter after deployment, you must manually update the tags on your VPCs.

Parameter	Default	Description
Tag key for Route Propagation to TGW Route Table(s)	Propagate-to	Specify a custom tag key name to initiate the route propagation to the transit gateway route table(s) workflow. NOTE: After initial deployment, don't change this solution's default parameter. If you change this parameter after deployment, you must manually update the tags on your VPCs.
(Optional) Comma separated list of VPC tag keys to copy from VPC to TGW Attachments	Associate-with, Propagate-to	Comma-separated list of tag keys (don't include Name). If the VPC has these tag keys, the tag key and value are copied to the created TGW attachment(s).
Transit Gateway Peering Tag	TgwPeer	Transit Gateway tag to monitor for peering connections. The tag value must follow the format <code>tgw-id_aws-region/tgw-id_aws-region</code> . For example, use <code>tgw-12345678_us-east-1/tgw-567890123_us-east-2</code> to create peering attachments with the two peers. You can update the value at any time.
Notification Settings		

Parameter	Default	Description
Receive Approval Notifications	No	Choose whether to receive approval notifications.
Approval Notification Email	<Optional input>	The email address for approval notifications. To use this parameter, you must set the Receive Approval Notifications parameter to Yes.

5. Choose **Next**.
6. On the **Configure stack options** page, choose **Next**.
7. On the **Review and create** page, review and confirm the settings. Choose the box acknowledging that the template creates IAM resources.
8. Choose **Submit** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should see a status of **CREATE_COMPLETE** in approximately 25 minutes.

After the stack is created, you receive two emails that contain temporary passwords for the read-only user and the admin user. If you enabled approval notification, Amazon SNS sends a subscription confirmation email with a link to the solution's web UI. You can also find the link to the web UI in the CloudFormation stack **Outputs** tab. The link is the **Value** of the **Console URL**. The system-generated password must be changed the first time you sign in.

Note

The temporary account expires if you don't sign in within seven days. Your new password must be at least 10 characters long.

Step 4: Launch the spoke stack(s)

Follow the step-by-step instructions in this section to configure and deploy the spoke stack(s) into your account(s).

Note

You must wait for the hub stack deployment to complete before you launch the spoke templates. The spoke templates depend on the EventBridge rule created during the hub stack launch. Additionally, deploy all templates in the same Region.

1. Sign in to your AWS spoke account using the AWS Management Console and select the button to launch the `network-orchestration-spoke.template` CloudFormation template.

Launch solution

2. Launch this template in the same Region as the hub template. The template launches in the US East (N. Virginia) Region by default.
3. On the **Create stack** page, verify that the correct template URL shows in the **Amazon S3 URL** text box and choose **Next**.
4. On the **Specify stack details** page, assign a name to your solution stack For information about naming character limitations, see [IAM and AWS STS quotas](#) in the *AWS Identity and Access Management User Guide*.
5. For **Parameters**, review the parameters for the template and modify them as necessary. This stack uses the following default values.

Parameter	Default	Description
Account ID of the network account where Transit Gateway resides.		
Network (Hub) Account	<i><Requires input></i>	The account ID for the hub account.

6. Choose **Next**.
7. On the **Configure stack options** page, choose **Next**.
8. On the **Review and create** page, review and confirm the settings. Choose the box acknowledging that the template creates IAM resources.
9. Choose **Submit** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should see a status of **CREATE_COMPLETE** in approximately three to four minutes.

Step 5: Add tags

Follow the step-by-step instructions in this section to add tags to your VPCs and subnets to create or update transit gateway attachments to the VPC or to your transit gateway to create transit gateway peering attachments. Changing tag keys and values (if applicable) results in identifying the transit gateway route tables to create associations and enable propagations. Deleting the tags results in deleting the resources created when the tag was added.

For a real-world scenario on how to configure tag values with this solution, refer to [Implementing Serverless Transit Network Orchestrator \(STNO\) in AWS Control Tower](#). For information on using [Organizations tag policies](#) with this solution, refer to [Enforce compliance using AWS Organizations tag policies with Serverless Transit Network Orchestrator \(STNO\)](#).

Transit gateway attachments to a VPC

This section provides instructions for attaching VPCs to your transit gateway.

Add tags to VPCs

Follow the step-by-step instructions in this section to add tags to your VPCs.

1. Sign in to your spoke account.
2. Navigate to the [Amazon VPC console](#).
3. Choose **VPCs**.
4. Select **Tags** and choose **Manage tags**.
5. Choose **Add new tag**.
6. Add the key-value pairs listed in [VPC tags](#).

Note

If you're using custom names for your transit gateway route tables, the values of the tags you assign need to match the names of the route tables associated with the transit gateway you're connecting to.

Add tags to subnets

Follow the step-by-step instructions in this section to add tags to your subnets.

1. Sign in to your spoke account.
2. Navigate to [Subnets](#) within the Amazon VPC console.
3. Select the subnet that you want to attach to the transit gateway.
4. Select **Tags** and choose **Manage tags**.
5. Choose **Add new tag**.
6. Add keys (without values) listed in [Subnet tags](#).

Add transit gateway attachments

Tags identify applicable resources, such as VPCs and subnets, in your spoke accounts. Tags allow create, read, update, and delete (CRUD) operations to run on the transit gateway route table associations and propagation.

Note

Verify that you have the appropriate access privileges to tag VPCs in spoke accounts, or identify the appropriate administrator in your organization.

VPC tags

For this solution to manage the VPC, the VPC in the spoke account must be tagged with both the **Associate-with** and **Propagate-to** keys. You must also add or remove both keys at the same time. By default, the tags are configured for automatic approval.

Key	Value	Description
Associate-with	<i><Requires input></i>	The default key is Associate-with . The value can be one of the default route table names (Flat, Isolated, Infrastructure , or On-premises) or a custom key.

Key	Value	Description
		You can change the name of the key in the template during initial configuration, but you must use the same key name when you tag the VPC. Type: String For sample route table options, refer to Custom route tables.
Propagate-to	<i><Requires input></i>	The default key is Propagate-to . The value can be one or more default route table names (Flat, Isolated, Infrastructure , or On-premises) or a key(s) that you created. You can change the name of the key in the template during initial configuration, but you must use the same key name when you tag the VPC. Type: CommaDelimitedList For sample route table options, refer to Custom route tables.

Subnet tags

Note

For a transit gateway attachment to a VPC, you can add only one subnet per Availability Zone. You can't attach a second subnet in the same Availability Zone to the transit gateway. Starting in version 3.3.0 of this solution, we support a new tag key **Route-to-tgw** that skips adding the subnet in the transit gateway attachment and only updates the associated route table with the default route.

Key	Value	Description
Attach-to-tgw	<i><Leave blank></i>	The default key is <code>Attach-to-tgw</code>.  Important Don't enter a value. You can change the name of the key in the template during initial configuration, but you must use the same key name when you tag the subnet. If there isn't an explicit route table associated with the subnet, the solution updates the main route table with the default route.
Route-to-tgw	<i><Leave blank></i>	The default key is <code>Route-to-tgw</code>.

Key	Value	Description
		 Important Don't enter a value. You can change the name of the key in the template during initial configuration, but you must use the same key name when you tag the subnet. If there isn't an explicit route table associated with the subnet, the solution updates the main route table with the default route.

Transit gateway peering attachments

This section provides instructions for creating transit gateway peering attachments.

See [Add or edit tags for a transit gateway](#) for instructions on adding and editing tags.

Add tags to transit gateway

Key	Value	Description
TgwPeer	<i><tgw-id_aws_region></i>	The default key prefix is TgwPeer. The default delimiter in the tag value to list multiple transit gateway per region is slash(/).

Key	Value	Description
		You must provide a valid Region name where the remote transit gateway exists. The value must be the remote transit gateway ID that exists in the <code><aws_region></code> that you provided in the key. For example, if you create a peering attachment with a transit gateway (tgw-2222) in the U.S. East (Ohio) Region (us-east-2), the tag key would be TgwPeer and the value would be tgw-2222_us-east-2 . You can peer with a second transit gateway in the same Region by using the same delimiter to add another suffix the tag key. In this case, the tag key would be TgwPeer and the value must be tgw-2222_us-east-2 /tgw-3333_us-east-1 . You can change the Transit Gateway Peering Tag add reference to the CFN parameter used in the key in the template during initial configuration, but you must use the same key name when you tag the transit gateway.

Key	Value	Description
		 Note Changing the value doesn't update the attachment. You must delete the tag and recreate it with a new value, then recreate the peering attachment.

Note

If you created peering attachments using solution versions before v3.0.0 and then upgraded to v3.0.0 or later, those attachments cannot be deleted by updating the tag value on the transit gateway. You will need to delete them manually.

Important

To route traffic between peered transit gateways, you must add a static route to the selected transit gateway route table that points to the transit gateway peering attachment. After you create the route, associate the same transit gateway route table with the transit gateway peering attachment. Refer to [Create a static route](#) for more information.

Custom compliance for network changes

This section provides instructions for custom compliance.

Add tags to transit gateway route table

Each transit gateway route table is tagged with an **ApprovalRequired** tag key with a default value of No. You can set the value to:

- Yes to enforce manual approval
- Conditional and add custom rules for compliance

Key	Value	Description
ApprovalRequired	No	The default value is No. This default setting allows any auto-approved or manually approved association and propagation changes.
ApprovalRequired	Yes	This setting enforces the manually approved workflow for any change in the association and propagation changes.
ApprovalRequired	Conditional	You can use this setting to automate approving or rejecting requests separately for associations and propagations. You can also optionally define rules based on the requesting account's OU.

Custom compliance rules

Administrators can change from the default automatic approval setup to manual approval by changing the **ApprovalRequired** tag value for every transit gateway route table individually.

See [View transit gateway route tables](#) for instructions on viewing your transit gateway route tables and updating tags.

The following tag keys and values are required with at least one rule if the **ApprovalRequired** is set to Conditional.

Key	Value	Description
ApprovalRule-Default-Association	Reject Accept ApprovalRequired	Default approval action for Associate-with route tables if none of the custom rules match. Enter Reject, Accept, or ApprovalRequired to match your desired action.
ApprovalRule-Default-Propagation	Reject Accept ApprovalRequired	Default approval action for Propagate-to route tables if none of the custom rules match. Enter Reject, Accept, or ApprovalRequired to match your desired action.
ApprovalRule-<i><NN></i>-InOUs ApprovalRule-<i><NN></i>-NotInOUs	Root/OUName1, Root/OUName2	A comma-separated list of OU paths starting with Root/. If you enter the key with the InOUs string, the rule checks if the account is in one of these OUs. If you enter the key with the NotInOUs string, the rule checks if the account isn't in any of the specified OUs.  Note <i><NN></i> denotes a two-digit number 01-99. Review service quotas for the tags for each

Key	Value	Description
		resource in your account.
ApprovalRule--<NN>-Association	Reject Accept ApprovalRequired	The approval action to take for a VPC that associates with this route table if the ApprovalRule- <NN>-In0Us or ApprovalRule- <NN>-NotIn0Us check matches. Enter Reject, Accept, or ApprovalRequired to match your desired action.  Note <NN> denotes a two-digit number 01-99. Review service quotas for the tags for each resource in your account.

Key	Value	Description
ApprovalRule--<NN>-Propagation	Reject Accept ApprovalRequired	The approval action to take for a VPC that propagates to this route table if the ApprovalRule- <NN>-InOUs or ApprovalRule- <NN>-NotInOUs check matches. Enter Reject, Accept, or ApprovalRequired to match your desired action.  Note <NN> denotes a two-digit number 01-99. Review service quotas for the tags for each resource in your account.

 Note

If you don't provide a value for the **ApprovalRule** keys, the default value is ApprovalRequired_.

Example: Infrastructure route table rules using OU membership

If your VPCs provide organization-wide shared services, such as Microsoft Active Directory and patching servers, and are limited to AWS accounts in the Infrastructure or Security OU, you can use the following rules to ensure that only VPCs in those OUs associate with the Infrastructure route table without approval. This prevents workload VPCs accidentally associating with the Infrastructure route table, which could inadvertently expose them to the entire organization.

This example also demonstrates how you can prevent VPCs in Sandbox OUs from accessing the organizational shared services. Together, the following rules auto-reject associations or propagations from Sandbox VPCs for the Infrastructure route table.

Infrastructure Route Table Tag Key	Value
Name	Infrastructure
ApprovalRequired	Conditional
ApprovalRule-Default-Association	ApprovalRequired
ApprovalRule-Default-Propagation	ApprovalRequired
ApprovalRule-01-InOUs	Root/Infrastructure/, Root/Security/
ApprovalRule-01-Association	Accept
ApprovalRule-01-Propagation	Accept
ApprovalRule-02-InOUs	Root/Sandbox/
ApprovalRule-02-Association	Reject
ApprovalRule-02-Propagation	Reject

Update the solution

If you have previously deployed the solution, follow these procedures to update the stacks to get the latest version of the solution's framework.

Important

To successfully update the hub stack from an earlier version of this solution to version 3.3.0 or later, provide input for the two required parameters for the hub stack: **Cognito Domain Prefix** and **Allow Listed Ranges**.

Update the organization role stack (optional)

Follow the step-by-step instructions in this section to update the organization role stack for your Organizations management account.

1. Sign in to the [AWS CloudFormation console](#), select your existing Network Orchestration for AWS Transit Gateway CloudFormation stack, and select **Update**.

Note

This solution was previously called Serverless Transit Network Orchestrator.

2. Select **Replace current template**.
3. Under **Specify template**:
 - a. Select **Amazon S3 URL**.
 - b. Copy the link of the `network-orchestration-organization-role.template` [CloudFormation template](#).
 - c. Paste the link in the **Amazon S3 URL** box.
 - d. Verify that the correct template URL shows in the **Amazon S3 URL** text box, and choose **Next**. Choose **Next** again.
4. Under **Parameters**, review the parameters for the template and modify them as necessary. For details about the parameters, see [Step 1: Launch the organization role stack \(optional\)](#).
5. Choose **Next**.
6. On the **Configure stack options** page, choose **Next**.

7. On the **Review** page, review and confirm the settings. Choose the box acknowledging that the template creates IAM resources.
8. Choose **View change set** and verify the changes.
9. Choose **Update stack** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive a `UPDATE_COMPLETE` status in approximately three to four minutes.

Update the hub stack(s)

Follow the step-by-step instructions in this section to update the hub stack(s).

Hub stack

Important

To mitigate the `AlreadyExists` error code for the service-linked role (**`AWSServiceRoleForResourceAccessManager`**) in the hub stack, we moved the role to a separate stack for v3.3.1 of this solution. This way, you can upgrade the hub stack in multiple Regions. You must deploy the service-linked role for AWS RAM hub stack *after* updating the hub stack.

Follow the step-by-step instructions in this section to update the hub stack.

1. Sign in to the [AWS CloudFormation console](#), select your existing Network Orchestration for AWS Transit Gateway CloudFormation stack, and select **Update**.

Note

This solution was previously called Serverless Transit Network Orchestrator.

2. Select **Replace current template**.
3. Under **Specify template**:
 - a. Select **Amazon S3 URL**.
 - b. Copy the link of the `network-orchestration-hub.template` [CloudFormation template](#).
 - c. Paste the link in the **Amazon S3 URL** box.

- d. Verify that the correct template URL shows in the **Amazon S3 URL** text box, and choose **Next**. Choose **Next** again.
4. Under **Parameters**, review the parameters for the template and modify them as necessary. For details about the parameters, see [Step 3: Launch the hub stack \(optional\)](#).

Important

To successfully upgrade from an earlier version of this solution to version 3.3.0 or later, provide input for the two required parameters for the hub stack: **Cognito Domain Prefix** and **Allow Listed Ranges**.

5. Choose **Next**.
6. On the **Configure stack options** page, choose **Next**.
7. On the **Review** page, review and confirm the settings. Choose the box acknowledging that the template creates IAM resources.
8. Choose **View change set** and verify the changes.
9. Choose **Update stack** to deploy the stack.

You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive a `UPDATE_COMPLETE` status in approximately 25 minutes.

Service-linked role for AWS RAM hub stack

The hub stack deletes the **AWSServiceRoleForResourceAccessManager** service-linked role. Re-deploy the service-linked role using [Step 2](#) of **Deploy the solution**.

Update the spoke stack(s)

Follow the step-by-step instructions in this section to update the spoke stacks.

Spoke stack

Important

The service-linked role for Transit Gateway (**AWSServiceRoleForVPCTransitGateway**) has been consolidated into the spoke stack. During the stack update, a custom resource

checks whether the role already exists. If the role exists, creation is skipped. If the role does not exist, it is created automatically.

Follow the step-by-step instructions in this section to update the spoke stack(s) for your spoke account(s).

1. Sign in to the [AWS CloudFormation console](#), select your existing Network Orchestration for AWS Transit Gateway CloudFormation stack, and select **Update**.

 Note

This solution was previously called Serverless Transit Network Orchestrator.

2. Select **Replace current template**.
3. Under **Specify template**:
 - a. Select **Amazon S3 URL**.
 - b. Copy the link of the `network-orchestration-spoke.template` [CloudFormation template](#).
 - c. Paste the link in the **Amazon S3 URL** box.
 - d. Verify that the correct template URL shows in the **Amazon S3 URL** text box, and choose **Next**. Choose **Next** again.
4. Under **Parameters**, review the parameters for the template and modify them as necessary. For details about the parameters, see [Step 4: Launch the spoke stack\(s\)](#).
5. Choose **Next**.
6. On the **Configure stack options** page, choose **Next**.
7. On the **Review** page, review and confirm the settings. Choose the box acknowledging that the template creates IAM resources.
8. Choose **View change set** and verify the changes.
9. Choose **Update stack** to deploy the stack.

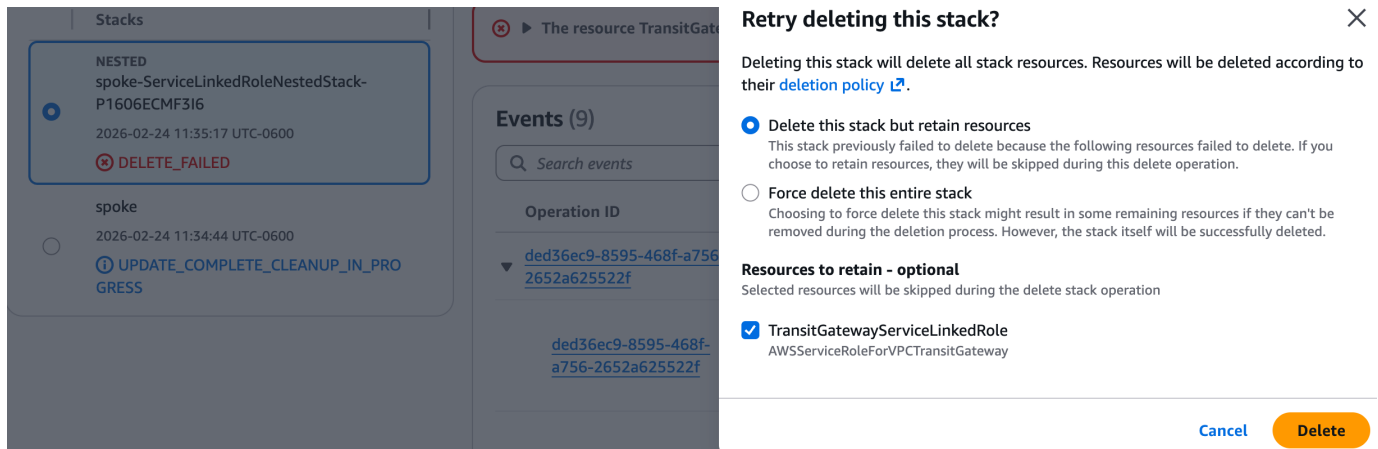
You can view the status of the stack in the AWS CloudFormation console in the **Status** column. You should receive a `UPDATE_COMPLETE` status in approximately three to four.

Upgrade from nested stack to consolidated spoke stack.

If you are upgrading to the latest version from any version between 3.3.11 and 3.3.22, follow these steps to migrate the spoke template from a nested stack to a consolidated spoke stack.

1. Delete the spoke stack.

- If the role is in use, the deletion fails. Select **Retain resource** for `TransitGatewayServiceLinkedRole` and delete the stack.



- If the role is not in use, the stack deletion cleans up the role.

2. Deploy the spoke template with the latest version. For details, see [Step 4: Launch the spoke stack\(s\)](#).

Migrate the hub stack (optional)

Follow the step-by-step instructions in this section to migrate the hub stack from earlier versions to the latest version.

Migrate from v1 to v3

Important

You cannot directly update solution v1 to latest due to breaking changes in latest, including `UserPoolAddOns` which is not supported under the Essential pricing tier.

Hub template

1. Download the latest version of the hub template from the link of the network-orchestration-hub.template [CloudFormation template](#).
2. Make the following change in UserPool Attribute Name in the hub template by removing custom: from the attribute name. Cognito is adding custom to the attribute so we don't need to put it in our configuration. For more information, see [SchemaAttribute](#) in the AWS CloudFormation User Guide.

Edit the template to remove the custom: prefix from the Cognito User Pool attribute:

```
# Change from:  
Name: custom:CognitoUserGr  
  
# To:  
Name: CognitoUserGr
```

3. Save the changes to the hub template.
4. Sign in to the [Amazon Cognito console](#), select your user pool created by v1 hub, and navigate to **Feature plan** under settings. Select **Plus** plan instead of **Essential** plan.
5. Follow the steps provided in [the section called "Hub stack"](#) to update the hub stack using the modified hub template.

Spoke template

For spoke template, follow the steps provided in [the section called "Spoke stack"](#).

Migrate from v2 to v3

Hub template

1. Download the latest version of the hub template from the link of the network-orchestration-hub.template [CloudFormation template](#).
2. Make the following change in UserPool Attribute Name in the hub template by removing custom: from the attribute name. Cognito is adding custom to the attribute so we don't need to put it in our configuration.

Edit the template to remove the custom: prefix from the Cognito User Pool attribute:

```
# Change from:  
Name: custom:CognitoUserGr  
  
# To:  
Name: CognitoUserGr
```

3. Save the changes to the hub template.
4. Follow the steps provided in [the section called "Hub stack"](#) to update the hub stack using the modified hub template.

Spoke template

For spoke template, follow the steps provided in [the section called "Spoke stack"](#).

Note

If you created peering attachments using solution versions before v3.0.0 and then upgraded to v3.0.0 or later, those attachments cannot be deleted by updating the tag value on the transit gateway. You will need to delete them manually.

Troubleshooting

This solution logs errors, warnings, informational messages, and debugging messages for the solution's Lambda functions. To choose the type of messages to log:

1. Find the applicable function in the [AWS Lambda console](#).
2. Change the **LOG_LEVEL** environment variable to the applicable type of message.

Level	Description
ERROR	Logs include information about anything that causes an operation to fail.
WARNING	Logs include information about anything that can potentially cause inconsistencies in the function but not necessarily cause the operation to fail. Logs also include ERROR messages.
INFO	Logs include high-level information about how the function is operating. Logs also include ERROR and WARNING messages.
DEBUG	Logs include information that might be helpful when debugging a problem with the function. Logs also include ERROR, WARNING, and INFO messages.

If these instructions don't address your issue, [Contact AWS Support](#) provides instructions for opening an AWS Support case for this solution.

Problem: Amazon EventBridge event bus permissions error

During spoke stack deployment, if you receive a **CREATE_FAILED** message for the 0 or the 1, verify that the Amazon EventBridge event bus in the hub account allows the spoke account to send events to the primary account.

Resolution

Update the hub stack with the spoke account ID or follow the [Managing event bus permissions](#) instructions to enter the applicable spoke account ID for **Principal**.

Problem: AlreadyExists error

During hub or spoke stack deployment, you receive an `0` error code. The following is an example.

```
Service role name AWSServiceRoleForVPCTransitGateway has been taken in this account,
please try a different suffix. (Service: Iam, Status Code: 400, Request ID: <request-
id>) (RequestToken: <request-token>, HandlerErrorCode: AlreadyExists)
```

Resolution

If you're deploying a new stack, review the instructions in [Step 2](#) of **Deploy the solution**.

If you're updating an existing stack, review the instructions in [Update the hub stack\(s\)](#) and [Update the spoke stack\(s\)](#).

Contact AWS Support

If you have [AWS Business Support+](#), [AWS Enterprise Support](#), or [Unified Operations](#), you can use the AWS Support Center to get expert assistance with this solution. The following sections provide instructions.

Create case

1. Sign in to [Support Center](#).
2. Choose **Create case**.

How can we help?

1. Choose **Technical**.
2. For **Service**, select **Solutions**.
3. For **Category**, select **Other Solutions**.

4. For **Severity**, select the option that best matches your use case.
5. When you enter the **Service**, **Category**, and **Severity**, the interface populates links to common troubleshooting questions. If you can't resolve your question with these links, choose **Next step: Additional information**.

Additional information

1. For **Subject**, enter text summarizing your question or issue.
2. For **Description**, describe the issue in detail, including the name of this solution and the version you are using, such as this example: **Network Orchestration for AWS Transit Gateway vX.Y.Z**.
3. Choose **Attach files**.
4. Attach the information that AWS Support needs to process the request.

Help us resolve your case faster

1. Enter the requested information.
2. Choose **Next step: Solve now or contact us**.

Solve now or contact us

1. Review the **Solve now** solutions.
2. If you can't resolve your issue with these solutions, choose **Contact us**, enter the requested information, and choose **Submit**.

Uninstall the solution

You can uninstall the solution from the AWS Management Console or by using the AWS CLI. You must manually delete the transit gateway, Amazon S3 buckets, and global network created by this solution. AWS solutions don't automatically delete these resources, in case you have stored data that you want to retain.

Using the AWS Management Console

1. Sign in to the [AWS CloudFormation console](#).
2. On the **Stacks** page, select this solution's installation stack.
3. Choose **Delete**.

Using AWS Command Line Interface

Determine whether the AWS CLI is available in your environment. For installation instructions, see [What is the AWS Command Line Interface?](#) in the *AWS CLI User Guide*. After confirming that the AWS CLI is available, run the following command.

```
$ aws cloudformation delete-stack --stack-name <installation-stack-name>
```

Manually delete resources

We configured this solution to retain solution-specific resources to prevent accidental data loss if you decide to delete the CloudFormation stack. After uninstalling the solution, you can manually delete these resources if you don't need to retain the data. Refer to these guides to delete the following resources:

- [Delete a transit gateway](#)
- [Delete Amazon S3 buckets](#)
- [Delete an AWS Network Manager global network](#)

Use the solution

This section provides a user guide for the solution's web UI and instructions to [using and customizing route tables](#).

Use the web UI

Important

If you don't deploy the UI, you can't approve or reject a network change. All the network changes will be auto-approved. You can use the compliance rules to auto-approve and auto-reject network changes.

Sign in to the web UI

After the hub stack is successfully deployed, you receive two emails containing a link to the web UI and sign-in credentials. By default, the solution creates one Amazon Cognito `adminuser` (in the admin group) and one Amazon Cognito `readonlyuser` (in the read-only group). For more information, refer to [Managing and searching for user accounts](#) in the *Amazon Cognito Developer Guide*.

Note

If you configured an external SAML-based identity provider in [step 3 \(SAML Provider Name parameter\)](#), instead of signing in with sign-in credentials, you can choose the button that redirects to your identity provider's sign in page. On the first sign in, the solution automatically adds every user to the **ReadOnlyUserGroup** and thereby grants them read access to the web UI. After a user signs in with read access, you can assign them to the **AdminGroup** with Amazon Cognito if needed. For more information, see [Adding groups to a user pool](#) in the *Amazon Cognito Developer Guide*.

Follow the step-by-step instructions in this section to sign in to the web UI.

1. Choose the link to open the web UI.

2. Enter the provided user credentials to sign in. You must change the system-generated password the first time that you sign in.

Note

The temporary account expires if you don't sign in within seven days. Your new password must be at least ten characters long.

Manage network activities

You can use the web UI to access the dashboard to view network changes, access action items to view, approve or reject network requests when manual approval is required, and view the history of all changes made within the solution.

Note

Information and history for a VPC are set to expire based on the time you specify in the hub template at stack launch. The default time is 90 days. Expired requests are automatically deleted from DynamoDB within 48 hours and are not shown in the web UI after deletion.

Access the dashboard

The **Dashboard** tab displays fields containing information about network changes stored in DynamoDB such as **VPC ID**, **VPC CIDR**, **Status**, **Association Route Table**, **Propagation Route Tables**, **Spoke Account**, **Subnet ID**, **Availability Zone**, and other relevant information. You can sort by these fields. You can also view the **Status** of each network change, including whether it was approved, rejected, auto-approved, or auto-rejected.

Access action items

The **Action Items** tab displays the requests that require [manual approval](#). If you chose to [automatically approve](#) requests, this tab will be empty. For manual approvals, each request contains the same fields as those in the **Dashboard** tab. Requests can have the following status: requested, processing, or failed. The reason for the failure displays in the comment column.

Approve or reject requests

When you enable [manual approval](#) for requests, the administrator approves or rejects the request using the web UI. Only users in the admin group can approve or reject requests. Users from the read only group can only view requests. When an administrator approves or rejects the request, the status is set to processing.

When a request is processing, users can't take further action from the web UI. The web UI calls a Lambda function, which initiates the solution state machine to process the request. After the process completes, state machine updates the request status, and the web UI reflects the new status.

View history of a request

To view the history of a request, select the request from either the **Dashboard** or **Action Items** tab and then choose **View History**.

Using and customizing route tables

This section provides a user guide for solution transit gateway route tables.

Default route tables

This solution creates the following default transit gateway route tables: Flat, Isolated, Infrastructure, and On-premises. Each route table and suggested propagations include a policy for common use cases.

- **Flat*route table** - VPCs associated with the Flat policy can reach other VPCs associated with the Flat, SharedServices, or Hybrid policies. The Flat policy enables a VPC to have connectivity to many other VPCs.
- **Isolated route table** - VPCs associated with the Isolated policy can reach VPCs with the SharedServices and Hybrid policies. VPCs in the Isolated policy can't use Transit Gateway to connect to other VPCs in the Isolated policy. This policy is for VPCs that don't communicate with each other.
- **Infrastructure route table** - VPCs associated with the SharedServices policy can reach other VPCs associated with the Isolated, Flat, or Hybrid policies. The SharedServices policy is used for VPCs that many other VPCs may rely on, such as shared authentication, shared tooling, or orchestration tools.

- **On-premises route table** - This route table is used for connecting to on-premises through either [AWS Virtual Private Network](#) (AWS VPN) or [AWS Direct Connect](#). Associate your on-premises connections to the On-premises route table.

Policy types	Associate with (route table name)	Propagate to (list of route table names)
Flat (east-west traffic)	Flat	Flat, On-premises, Infrastructure
Isolated (north-south traffic)	Isolated	On-premises, Infrastructure
SharedServices	Infrastructure	Flat, On-premises, Isolated
Hybrid	On-premises	Flat, Infrastructure, Isolated

Note

In this implementation guide, a policy is defined by both an association to a single transit gateway route table, and the transit gateway route table propagation. To implement these concepts, you must tag both the association and propagation on each spoke VPC according to the intended design. This is because the policies aren't centrally managed. Inconsistent tagging can create a drift between the desired policy and what is configured.

You can use the **ApprovalRequired** tag on route tables that need manual control. By default, we set up this solution for automatic approval, but you can change this tag to set up manual approval. See [Transit gateway route table tags](#) for more information about the **ApprovalRequired** tag.

For more information, refer to [On-premises connectivity](#).

Custom route tables

If the default policies don't meet your requirements, you can create your own transit gateway route table configurations.

Note

You don't need a separate route table for each VPC to achieve segmentation. Segmentation is accomplished by controlling the propagation. For example, an `Isolated` route table doesn't propagate to itself and, as a result, nothing associated with an `Isolated` route table is able to reach other `Isolated` resources through the transit gateway.

The administrator can create new transit gateway route tables in the Amazon VPC console in the hub account. The combination of route tables and propagation provided with the transit gateway allows for a variety of connection policies.

Example 1: Create a new custom route table

The following sample table and steps provide a guide for setting your routing policies. For this example, you implement a `Development` policy and route table. This policy and route table allow developers to create VPCs that don't have access to sensitive resources in the `Isolated` or `Infrastructure` route tables. You create a new transit gateway route table that propagates to the `Flat`, `On-premises`, and `Development` route tables.

Policy types	Associate with (route table name)	Propagate to (list of route table names)
Flat (east-west traffic)	Flat	Flat, On-premises, Infrastructure, Development
Isolated (north-south traffic)	Isolated	On-premises, Infrastructure
SharedServices	Infrastructure	Flat, On-premises, Isolated

Policy types	Associate with (route table name)	Propagate to (list of route table names)
Hybrid	On-premises	Flat, Infrastructure, Isolated, Development
Development	Development	Development, Flat, On-premises

Create the route table

1. In the hub account, sign in to the [Amazon VPC console](#).
2. In the navigation pane, choose **Transit gateway route tables**.
3. Choose **Create transit gateway route table**.
4. For **Name tag**, enter a name for the route table. For this example, enter Development.
5. For **Transit gateway ID**, select the appropriate transit gateway.
6. Choose **Create transit gateway route table**. You will receive a confirmation message.
7. Select **Close**.
8. Optional: If you want changes to this route table to be manually approved:
 - a. Select the newly created route table from the list in step 2.
 - b. Choose the **Tags** tab.
 - c. Choose **Add/Edit tags** and then choose **Create tag**.
 - d. In the **Key** field, enter ApprovalRequired and in the **Value** field, enter Yes.
 - e. Choose **Save**.

Determine the access model

After the new route table is created, determine the access model. For two transit gateway attachments to communicate, each of their associated route tables must have each other's routes.

1. Determine where your new route table should propagate routes, defined by the other transit gateway route tables. For example, should the Infrastructure route table be propagated from your new Development route table?

2. Check that the other route tables are reciprocating the propagation for two-way communication. For example, you may want to propagate Infrastructure, Flat, and On-premises route table to your new Development route table.

Note

If your custom route table requires access to VPCs that have already been attached, you must change the **Propagate-to** tag for each spoke VPC to include your new route table.

Associate VPCs

To associate a new VPC to this route table:

1. Tag the new VPC with the **Associate-with** key and reference the new route table name in the value. See [Step 5: Add tags](#) for detailed tagging instructions. For example:

```
Associate-with: <ExampleRouteTable>
```

2. Tag the new VPC with the **Propagate-to** key and reference the route tables you want to propagate to from the previous step. For example:

```
Propagate-to: Infrastructure, Flat, Hybrid
```

3. Tag one subnet in each Availability Zone. For example:

```
Attach-to-tgw: <leave blank>
```

Note

If you configured manual approval, you might need to sign in to the web UI to approve the change. If that is the case, you will receive an email with the request that contains the link to the web UI.

To confirm the attachment, look for attachments on the transit gateway in the VPC management console, in the web UI, or in the state machine history.

For more information about transit gateway route tables, refer to [Transit gateway route tables](#).

Example 2: Modify policies

If you don't need the provided Flat policy, you can modify the existing Flat policy to meet your custom requirements. Turn off propagation to the Infrastructure route table and remove the Flat propagation from the Infrastructure route table.

On-premises connectivity

This solution builds a base network, giving you automation to attach VPCs to a transit gateway. You can extend your on-premises network by creating transit gateway route tables using the web UI, creating VPN attachments, or attaching a transit gateway to an AWS Direct Connect gateway.

For instructions on how to manually attach a VPN to the transit gateway for on-premises connectivity, see [Transit gateway VPN attachments](#).

For instructions on how to manually attach Direct Connect to the transit gateway for on-premises connectivity, refer to [Transit gateway attachments to a Direct Connect gateway](#).

Developer guide

This section provides the source code for the solution.

Source code

This solution is coded in Python and the web UI is coded in ReactJS.

Visit our [GitHub repository](#) to download the source files for this solution and to share your customizations with others.

The [AWS CDK](#) generates the solution templates. See the [README.md](#) file for additional information.

Reference

This section includes information about data collection and a [list of builders](#) who contributed to this solution.

Data collection

This solution sends operational metrics to AWS (the "Data") about the use of this solution. We use this Data to better understand how customers use this solution and related services and products. AWS's collection of this Data is subject to the [AWS Privacy Notice](#).

Contributors

- Lalit Grover
- Abhinay Reddy
- Todd Gruet
- Alex Torres
- Rizvi Rahim
- Chaitanya Deolankar
- Garvit Singh
- Thiemo Belmega
- Ryan Garay

Revisions

Check the [CHANGELOG.md](#) file in the GitHub repository to see all notable changes and updates to the software. The changelog provides a clear record of improvements and fixes for each version.

Notices

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

The Network Orchestration for AWS Transit Gateway solution is licensed under the terms of the [Apache License Version 2.0](#).