**Title - AWS Landing Zone deployment using terraform**

*Project description :*

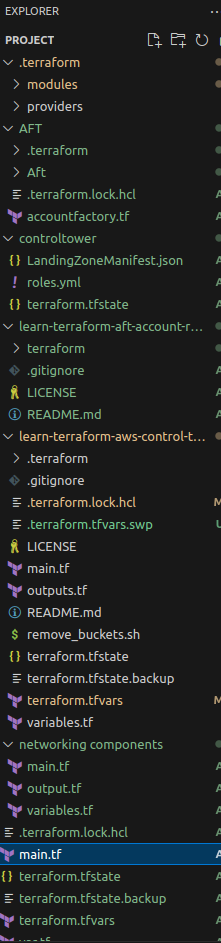
The purpose of this project is to deploy a standardized AWS Landing Zone infrastructure using Terraform, emphasizing best practices for security, networking, and account management. Leveraging tools such as AWS CLI/API and Terraform, the project encompasses the creation of an organized VPC architecture with secure network measures, automated account provisioning, and compliance monitoring.

*Tools Required:*

Terraform

AWS CLI

*Folder Structure :*



AWS Control Tower Account Factory for Terraform

AWS Control Tower Account Factory for Terraform (AFT) follows a GitOps model to automate the processes of account provisioning and account updating in AWS Control Tower. You'll create an \*account request\* Terraform file, which provides the necessary input that triggers the AFT workflow for account provisioning.

(https://docs.aws.amazon.com/controltower/latest/userguide/aft-overview.html)

Getting started

This guide is intended for administrators of AWS Control Tower environments who wish to set up Account Factory for Terraform (AFT) in their environment. It describes how to set up an Account Factory for Terraform (AFT) environment with a new, dedicated AFT management account.

This guide follows the deployment steps outlined in [Deploy AWS Control Tower Account Factory for Terraform (AFT)](https://docs.aws.amazon.com/controltower/latest/userguide/aft-getting-started.html)

## Configure and launch your AWS Control Tower Account Factory for Terraform

Five steps are required to configure and launch your AFT environment.

Step 1: Launch your AWS Control Tower landing zone

Before launching AFT, you must have a working AWS Control Tower landing zone in your AWS account. You will configure and launch AFT from the AWS Control Tower management account.

Step 2: Create a new organizational unit for AFT (recommended)

We recommend that you create a separate OU in your AWS Organization, where you will deploy the AFT management account. Create an OU through your AWS Control Tower management account. For instructions on how to create an OU, refer to Create an organization in the AWS Organizations User Guide.

Step 3: Provision the AFT management account

AFT requires a separate AWS account to manage and orchestrate its own requests. From the AWS Control Tower management account that's associated with your AWS Control Tower landing zone, you'll provision this account for AFT.

To provision the AFT management account, see [Provisioning Account Factory Accounts With AWS Service Catalog](https://docs.aws.amazon.com/controltower/latest/userguide/provision-as-end-user.html). When specifying an OU, be sure to select the OU you created in Step 2. When specifying a name, use "AFT-Management".

Note: It can take up to 30 minutes for the account to be fully provisioned. Validate that you have access to the AFT management account.

Step 4: Ensure that the Terraform environment is available for deployment

This step assumes that you are experienced with Terraform, and that you have procedures in place for executing Terraform. AFT supports Terraform Version 0.15.x or later.

Step 5: Call the Account Factory for Terraform module to deploy AFT

The Account Factory for Terraform module must be called while you are authenticated with AdministratorAccess credentials in your AWS Control Tower management account.

AWS Control Tower, through the AWS Control Tower management account, vends a Terraform module that establishes all infrastructure necessary to orchestrate your AWS Control Tower account factory requests. You can view that module in the AFT repository.

Next Steps:

Now that you have configured and deployed AWS Control Tower Account Factory for Terraform, follow the steps outlined in [Post-deployment steps](https://docs.aws.amazon.com/controltower/latest/userguide/aft-post-deployment.html) and [Provision accounts with AWS Control Tower Account Factory for Terraform](https://docs.aws.amazon.com/controltower/latest/userguide/taf-account-provisioning.html) to begin using your environment.

*Requirements*

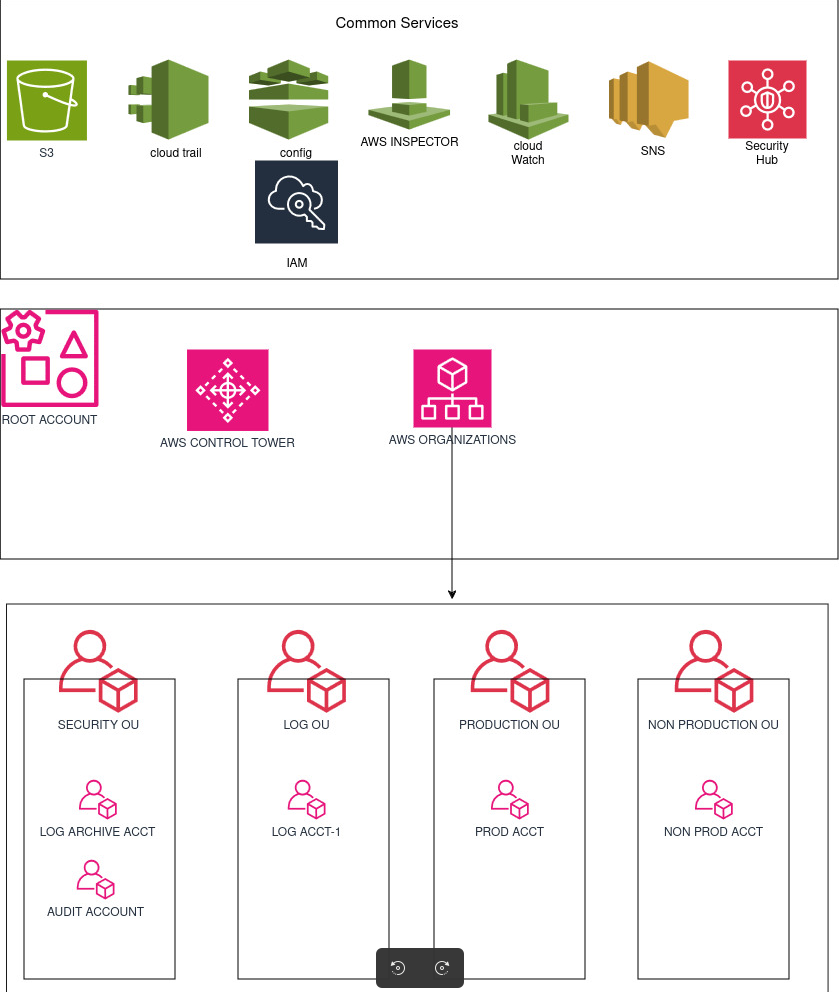
| Name | Version |

|------|---------|

| <a name="requirement\_terraform"></a> [terraform](#requirement\\_terraform) | >= 1.0.0, < 2.0.0 |

| <a name="requirement\_aws"></a> [aws](#requirement\\_aws) | >= 5.11.0, < 6.0.0 |

ARCHITECTURE DIAGRAM



OUTPUT

