**Task 2 - Terraform and Cloud: Container Hosting Infrastructure**

This repository contains the Terraform code to create infrastructure in AWS (or equivalent) for hosting a containerized application. The infrastructure setup can be either **server-based** or **serverless**. The infrastructure includes components like VPC, ECS/EKS, subnets, load balancers, and other necessary resources.

**Table of Contents**

1. [Overview](#overview)
2. [Infrastructure](#infrastructure)
   * [Server-based Setup](#server-based-setup)
   * [Serverless Setup](#serverless-setup)
3. [Getting Started](#getting-started)
4. [Prerequisites](#prerequisites)
5. [Usage](#usage)
6. [Push Code to Git Repository](#push-code-to-git-repository)
7. [Notes](#notes)
8. [License](#license)

**Overview**

This project sets up the necessary infrastructure to host a containerized application in the cloud using Terraform. Depending on the preference, either a server-based setup (using ECS/EKS) or a serverless setup (using Lambda) is created, with resources like VPC, subnets, and load balancers deployed accordingly.

**Infrastructure**

**Server-based Setup**

The infrastructure will be created as follows:

* **VPC** with 2 public and 2 private subnets.
* **ECS/EKS cluster** deployed within the VPC.
* **ECS/EKS task/service** running the container on private subnets.
* **Load Balancer** deployed on public subnets to offer the service.

**Serverless Setup**

The infrastructure will be created as follows:

* **VPC** with 2 public and 2 private subnets.
* **Lambda function** running the container.
* **API Gateway, CDN, or Load Balancer** to trigger the Lambda function.

**Getting Started**

To get started with this project, follow these steps:

1. Clone this repository:

bash

CopyEdit

git clone https://github.com/yourusername/terraform-container-infrastructure.git

cd terraform-container-infrastructure

1. Initialize Terraform to download the necessary providers:

bash

CopyEdit

terraform init

1. Review the Terraform plan to ensure the infrastructure is set up as desired:

bash

CopyEdit

terraform plan

1. Apply the Terraform configuration to create the infrastructure:

bash

CopyEdit

terraform apply

1. To destroy the resources (optional), run:

bash

CopyEdit

terraform destroy

**Prerequisites**

Before running the Terraform code, make sure you have the following prerequisites:

* **Terraform** installed on your machine (preferably the latest stable version).
* **AWS CLI** configured with appropriate credentials.
* Access to a public Git repository (e.g., GitHub, GitLab) for pushing the code.

**Usage**

After running terraform apply, your containerized application will be accessible based on the setup you have chosen (either ECS/EKS for server-based or Lambda for serverless). The infrastructure will include a load balancer (in the case of server-based) or an API Gateway (for serverless) that will trigger your container function.

To modify the container image or other settings, edit the relevant parts in the container\_definitions.json or equivalent files and re-apply the Terraform plan.