Terraform part

Mateusz Lewicki

July 11, 2020

Table of Contents

# Puprouse of using Terraform

## Brief of terrafom in current project

hhhhhh - [variables.tf](../terraform/variables.tf) - [images.tf](../terraform/images.tf) - [terraform.tfvars](../terraform/terraform.tfvars) - [terraform.tfstate](../terraform/terraform.tfstate) - [outputs.tf](../terraform/outputs.tf) - [networks.tf](../terraform/networks.tf) - [main.tf](../terraform/main.tf)

# File description

## main.tf (part of)

provider "docker" {  
}  
  
  
resource "docker\_container" "LB" {  
 image = docker\_image.lb.latest  
 name = "LB"  
 networks\_advanced{  
 name=docker\_network.public\_network.name  
 }  
 networks\_advanced{  
 name=docker\_network.app\_network\_1.name  
 }  
 networks\_advanced{  
 name=docker\_network.app\_network\_2.name  
 }  
 ports{  
 internal= var.http\_port  
 external= var.http\_port  
 }  
 ports{  
 internal = var.https\_port  
 external = var.https\_port  
 }  
 ports{  
 internal = 1936  
 external = 1936  
 }  
 host{  
 host="app1"  
 ip=docker\_container.apache\_1.ip\_address  
 }  
 host{  
 host="app2"  
 ip=docker\_container.apache\_2.ip\_address  
 }  
}  
  
[...]

## images.tf

resource "docker\_image" "alpine" {  
 name = "alpine:latest"  
 keep\_locally = true  
}  
  
resource "docker\_image" "web" {  
 name = "${var.registry}/${var.web\_image\_name}"  
 keep\_locally = true  
}  
  
resource "docker\_image" "db" {  
 name = "${var.registry}/${var.db\_image\_name}"  
 keep\_locally = true  
}  
  
resource "docker\_image" "lb" {  
 name = "${var.registry}/${var.lb\_image\_name}"  
 keep\_locally = true  
}

## networks.tf

resource "docker\_network" "app\_network\_1" {  
 name = "app\_network\_1"  
 internal = true  
 ipam\_config{  
 subnet="10.0.0.0/28"  
 }  
}  
  
resource "docker\_network" "app\_network\_2" {  
 name = "app\_network\_2"  
 internal = true  
 ipam\_config{  
 subnet="10.0.0.16/28"  
 }  
}  
resource "docker\_network" "public\_network" {  
 name = "public\_network"  
 ipam\_config{  
 subnet="10.0.0.32/28"  
 }  
}

## variables.tf

variable "http\_port" {  
 type = string  
}  
  
variable "https\_port" {  
 type = string  
}  
  
variable "db\_port" {  
 type = string  
}  
  
variable "registry" {  
 type = string  
}  
  
variable "db\_image\_name" {  
 type = string  
}  
variable "web\_image\_name" {  
 type = string  
}  
variable "lb\_image\_name" {  
 type = string  
}

## outputs.tf

output "apache\_1\_ip\_addr" {  
 value = docker\_container.apache\_1.ip\_address  
}  
  
output "apache\_2\_ip\_addr" {  
 value = docker\_container.apache\_2.ip\_address  
}  
output "LB\_ip\_addr" {  
 value = docker\_container.LB.ip\_address  
}

## terraform.tfvars

http\_port = "80"  
 https\_port = "443"  
 db\_port = "3306"  
 registry = "localhost:5000"  
 web\_image\_name = "lamp\_terr/web"  
 db\_image\_name = "lamp\_terr/database"  
 lb\_image\_name = "lamp\_terr/loadbalancer"