Terraform part

Mateusz Lewicki

July 11, 2020

Contents

Puprouse of using Terraform

Brief of terrafom in current project

- variables.tf
- images.tf
- terraform.tfvars
- terraform.tfstate
- outputs.tf
- networks.tf
- 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"
```