

Infraestructura como código

Por: Juan Contreras



Tabla de Contenido

module nivel_1 {

1.1 = “Infraestructura como código”

1.2 = “Terraform”

}

module nivel_2 {

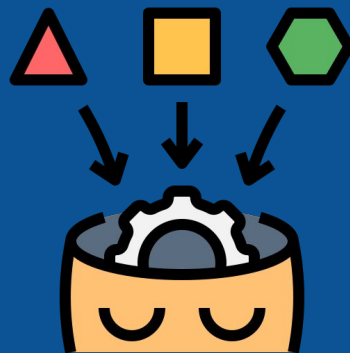
2.1 = “Extendiendo el poder de terraform”

2.2 = “Pruebas para generar confianza.”

2.3 = “Despliegue de la infraestructura con azure devops”

}

NIVEL 1



¿Qué es infraestructura como código?

Es una práctica implementada en la cultura devops, la cual implica escribir la configuración de la infraestructura deseada a través de codificación.



Ventajas

- Automatizar la creación de la infraestructura.
- Velocidad en los despliegues.
- Arquitectura consistente.
- Configuración repetible.
- Reducción de errores.



Herramientas de configuración de infraestructura

- Cloud formation.
- Pulumi.
- Ansible.
- Terraform.



¿Qué es Terraform?

Herramienta para la automatización de infraestructura mediante código que contiene un lenguaje declarativo y con un perfil multi nube.



Elementos de terraform

- Resource
- Data Source
- Module
- Provider
- State
- Backend
- Provider

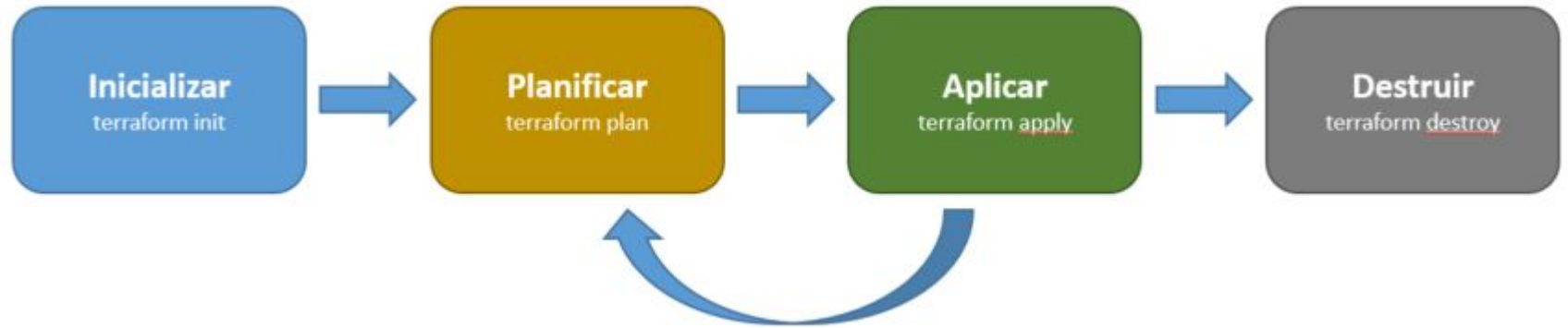


Comandos importantes

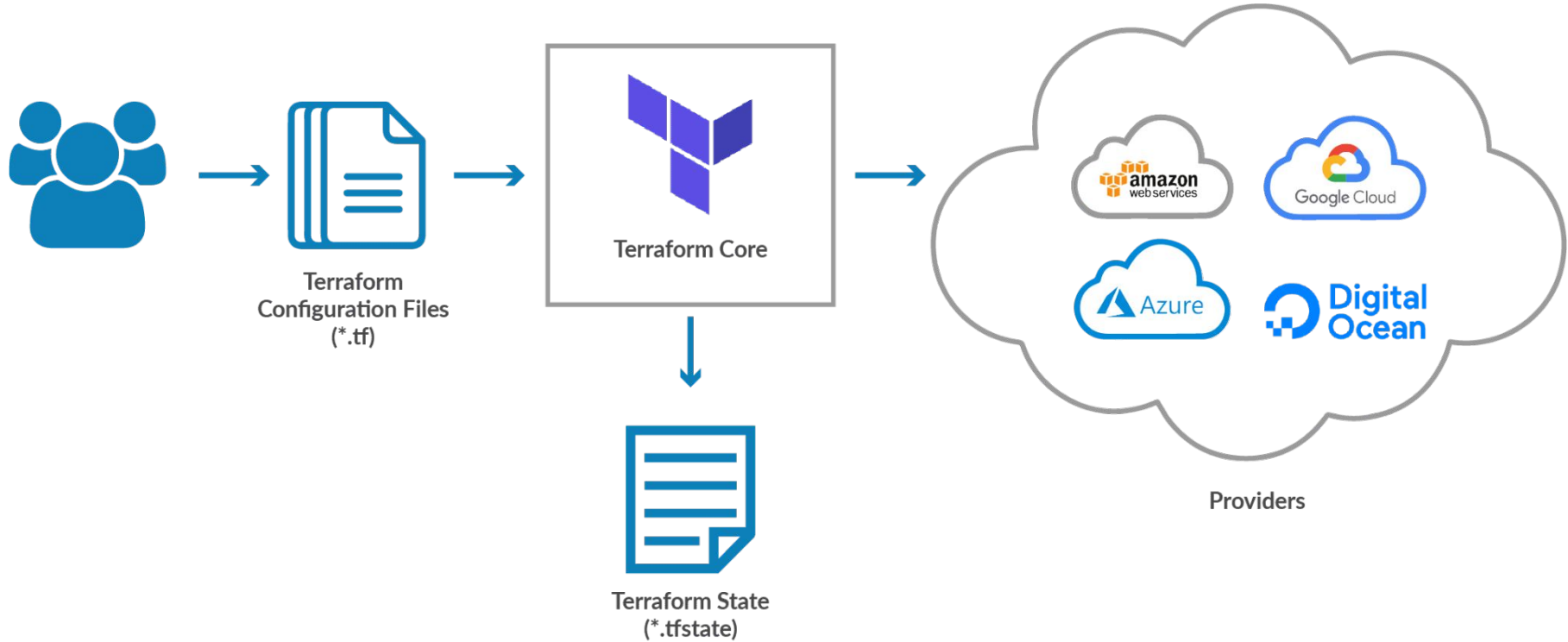
- terraform init
- terraform validate
- terraform plan
- terraform apply
- terraform destroy
- terraform graph
- terraform refresh
- terraform import



Ciclo de terraform



Terraform State



Resource

```
resource "<resource>" "<resource_name>" {  
  param_1 = "<value_1>"  
  param_2 = "<value_2>"  
}
```

```
resource "google_service_account" "service_account" {  
  account_id = "application"  
  display_name = "application"  
  description = "service account for the application."  
}
```

https://registry.terraform.io/providers/hashicorp/google/latest/docs/resources/google_service_account

Outputs

```
output "<output_name>" {  
  value = "<resource>.<resource_name>.<output_name>"  
}
```

```
output "id" {  
  value = google_service_account.service_account.id  
}  
  
output "name" {  
  value = google_service_account.service_account.name  
}  
  
output "unique_id" {  
  value = google_service_account.service_account.unique_id  
}  
  
output "email" {  
  value = google_service_account.service_account.email  
}
```

https://registry.terraform.io/providers/hashicorp/google/latest/docs/resources/google_service_account#attributes-reference

Variables

string

```
variable "name" {  
  description = "Name of the vpc to create."  
  type = string  
  default = ""  
}
```

number

```
variable "value" {  
  description = "Number of subnets to create."  
  type = number  
  default = 30  
}
```

bool

```
variable "is_private" {  
  description = "If the storage will be private or not."  
  type = bool  
  default = true  
}
```

list

```
variable "ip_range" {  
  description = "list of ip ranges."  
  type = list(string)  
  default = [  
    "10.0.0.0/16",  
    "10.1.0.0/16"  
  ]  
}
```

object

```
variable "ip_range" {  
  description = "list of ip ranges."  
  type = object({  
    name = string  
    address = string  
    prefix_length = number  
  })  
  default = {  
    name = "vpc-custom-default"  
    address = "10.1.0.0"  
    prefix_length = 22  
  }  
}
```

Uso de variables

```
resource "google_service_account" "service_account" {  
  account_id   = var.account_id  
  display_name = var.display_name  
  description  = var.description  
}
```

```
module "vpc" {  
  source = "terraform-google-modules/network/google"  
  version = "5.0.0"  
  
  project_id           = var.project_id  
  network_name         = var.network_name  
  routing_mode         = var.routing_mode  
  delete_default_internet_gateway_routes = var.delete_default_internet_gateway_routes  
  subnets             = var.subnets  
  secondary_ranges     = var.secondary_ranges  
  routes               = var.routes  
  firewall_rules       = var.firewall_rules  
}
```

Modules

```
module "<module_name>" {  
  source    = "<Puede ser local o remoto>"  
  version   = "<Versión estática del módulo>"  
  
  param_1 = "<value_1>"  
  param_2 = "<value_2>"  
}
```

```
resource "google_compute_global_address" "ip_block" {  
  count = length(var.service_ranges)  
  
  name       = var.service_ranges[count.index].name  
  purpose    = var.service_ranges[count.index].purpose  
  address_type = var.service_ranges[count.index].address_type  
  prefix_length = var.service_ranges[count.index].prefix_length  
  network    = var.network  
  address     = var.service_ranges[count.index].address  
}  
  
resource "google_service_networking_connection" "vpc_connection" {  
  network      = var.network  
  service      = var.service  
  reserved_peering_ranges = var.reserved_peering_ranges  
  
  depends_on = [  
    google_compute_global_address.ip_block  
  ]  
}
```

```
module "networking_connection" {  
  source = "../modules/networking_connection"  
  
  service_ranges = var.service_ranges  
  network        = var.network  
  service        = var.service  
  reserved_peering_ranges = var.reserved_peering_ranges  
}
```

<https://registry.terraform.io/modules/terraform-google-modules/network/google/latest/submodules/routes>

Outputs

```
output "<output_name>" {  
  value = "module.<module_name>.<output_name>"  
}
```

```
output "network" {  
  value = module.vpc.network  
}  
  
output "network_id" {  
  value = module.vpc.network_id  
}  
  
output "network_name" {  
  value = module.vpc.network_name  
}  
  
output "route_names" {  
  value = module.vpc.route_names  
}  
  
output "subnets_names" {  
  value = module.vpc.subnets_names  
}
```

NIVEL 2



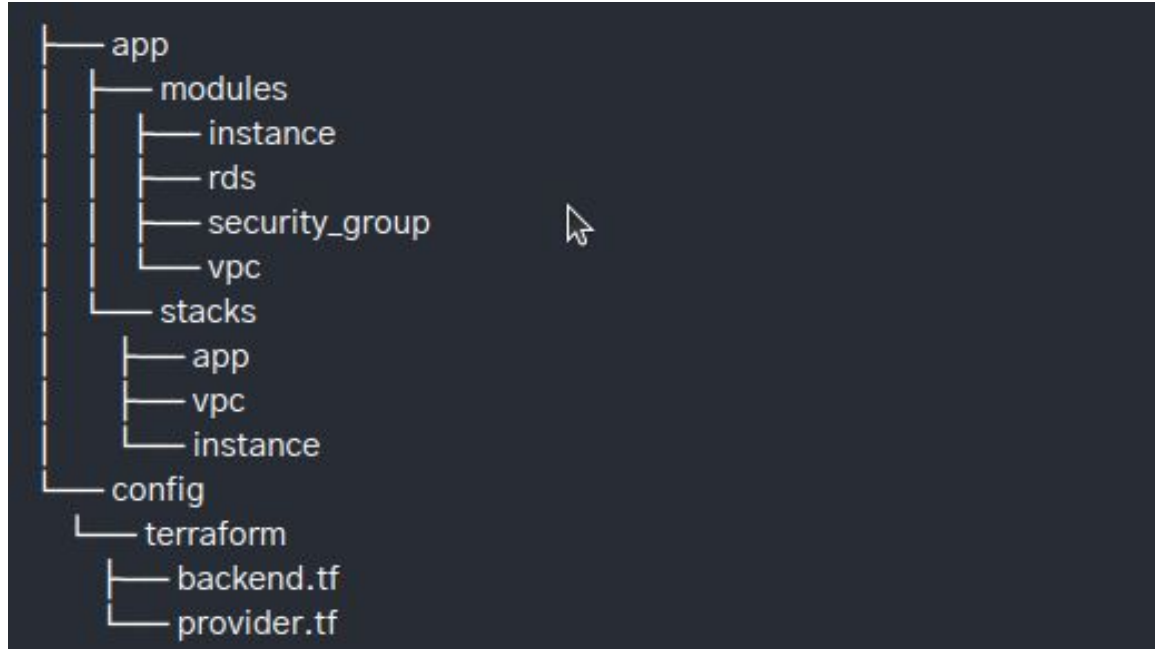
Extendiendo el poder de terraform

Es un framework de terraform,
que nos proporciona una
estructura organizada,
convenciones de configuración,
evita la duplicidad de código y
agrega herramientas de ayuda.



**TERRA
SPACE**

Estructura del proyecto



Código no repetido

base.tfvars

```
name          = "database-custom"  
database_version = "POSTGRES_14"  
disk_size     = 10  
disk_type     = "PD_HDD"
```

dev.tfvars

```
disk_size = 20
```

qa.tfvars

```
disk_size = 100  
disk_type = "PD_SSD"
```



Herramientas de ayuda

→ Commands

terraspace up
terraspace down
terraspace all up
terraspace all down
terraspace test
terraspace graph

→ Generators

terraspace new project <project_name>
terraspace new stack <stack_name>
terraspace new module <module_name>
terraspace new test <test_name>



Creación de bucket backend automática

