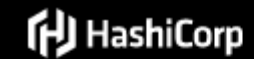




TERRAFORM CLOUD AUTOMATION TOOL



DevOps Training

@COPYRIGHT OF WWW.CLOUDBEARERS.COM

What is Terraform ?

Terraform is a tool for building, changing, and versioning infrastructure safely and efficiently. Terraform can manage existing and popular service providers as well as custom in-house solutions.

Configuration files describe to Terraform the components needed to run a single application or your entire datacenter. Terraform generates an execution plan describing what it will do to reach the desired state, and then executes it to build the described infrastructure. As the configuration changes, Terraform is able to determine what changed and create incremental execution plans which can be applied.

The infrastructure Terraform can manage includes low-level components such as compute instances, storage, and networking, as well as high-level components such as DNS entries, SaaS features, etc.

HASHICORP INFRASTRUCTURE AS CODE

Infrastructure is described using a high-level configuration syntax. This allows a blueprint of your datacenter to be versioned and treated as you would any other code. Additionally, infrastructure can be shared and re-used.

EXECUTION PLANS

Terraform has a "planning" step where it generates an *execution plan*. The execution plan shows what Terraform will do when you call apply. This lets you avoid any surprises when Terraform manipulates infrastructure.

TERRAFORM GRAPHS

Terraform builds a graph of all your resources, and parallelizes the creation and modification of any non-dependent resources. Because of this, Terraform builds infrastructure as efficiently as possible, and operators get insight into dependencies in their infrastructure.

CHANGE AUTOMATION

Complex change sets can be applied to your infrastructure with minimal human interaction. With the previously mentioned execution plan and resource graph, you know exactly what Terraform will change and in what order, avoiding many possible human errors.

TERRAFORM VS OTHER SOFTWARE

Terraform provides a flexible abstraction of resources and providers. This model allows for representing everything from physical hardware, virtual machines, and containers, to email and DNS providers. Because of this flexibility, Terraform can be used to solve many different problems. This means there are a number of existing tools that overlap with the capabilities of Terraform.

KEY ITEMS IN TERRAFORM

- build, change, version infrastructure through common config file
- codify everything from physical hardware, VMs, and containers, to email and DNS providers
- with multi-cloud/provider support!

HOW THAT WORK?

.tf config file allows teams to describe their infrastructure in simple DSL

terraform CLI creates, changes, and destroys these resources accordingly

Declare .tf file resources via HCL (HashiCorp Configuration Language):

WHAT'S THAT LOOK LIKE?

```
resource "digitalocean_droplet" "web" {  
  name = "tf-web"  
  size = "512mb"  
  image = "centos-5-8-x32"  
  region = "sfo1"  
}  
  
resource "dnsimple_record" "hello" {  
  domain = "example.com"  
  name = "test"  
  value = "${digitalocean_droplet.web.ipv4_address}"  
  type = "A"  
}
```

PARAMETERIZING CONFIG

Parameterizing our configuration

In the previous chapter we created some configuration in our `~/terraform/base/base.tf` configuration file.

Listing 3.1: Our original configuration

```
provider "aws" {
  access_key = "abc123"
  secret_key = "abc123"
  region     = "us-east-1"
}

resource "aws_instance" "base" {
  ami           = "ami-0d729a60"
  instance_type = "t2.micro"
}

resource "aws_eip" "base" {
  instance = aws_instance.base.id
  vpc      = true
}
```

PARTS OF VPC TERRAFORM CAN INPUT

PARTS OF A VPC

- ▶ VPC definition (CIDR block)
- ▶ Subnets
- ▶ Internet gateway to allow our subnet to route to the internet
- ▶ Routing table to define how traffic is routed
- ▶ Main route table association to tell the vpc what route table to use by default

TERRAFORM GRAPH

TERRAFORM WILL GRAPH DEPENDENCIES

▶ `terraform graph | dot -Tpng > graph.png`

it's
Q & A
TIME!



THANK YOU!

training@laksans.com

@copyright of www.cloudbearers.com