

Configure Resources After Creation

Automating Infrastructure Deployment



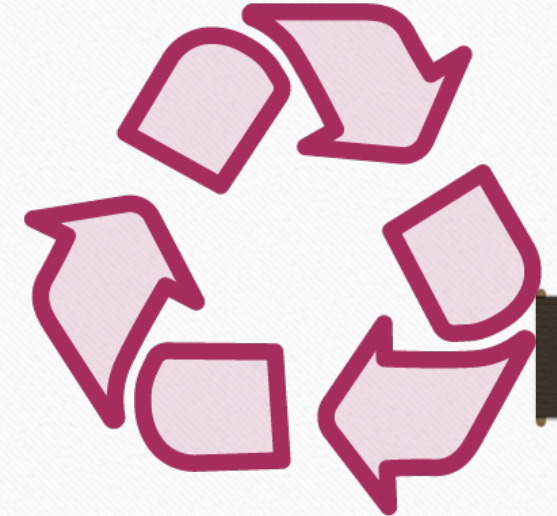
Provisioning
resources



Planning Updates

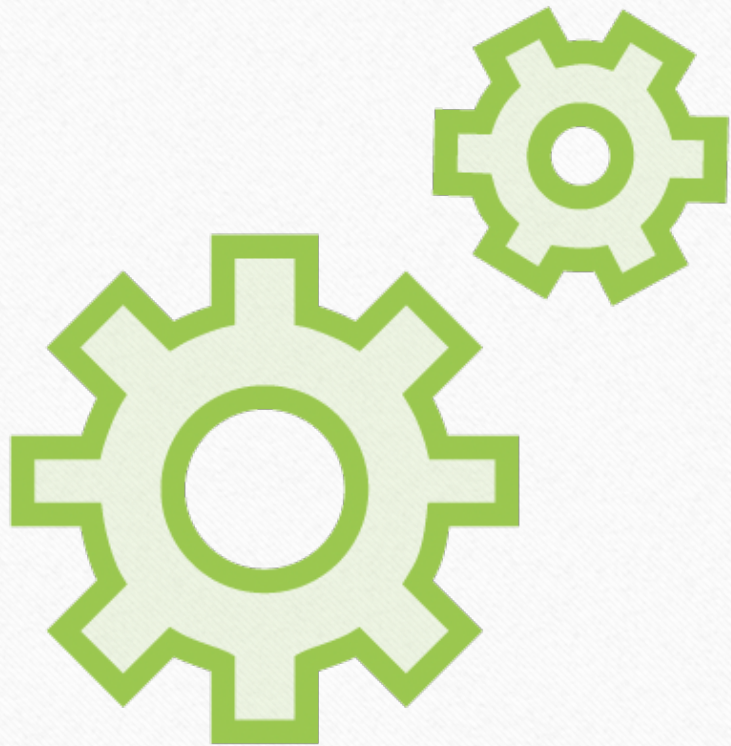


Using Source Control



Reusing templates

Terraform Syntax



HashiCorp configuration language

Why not JSON?

Human readable and editable

Configuration syntax and expressions

Conditionals, functions, templates

Terraform Syntax - Blocks

#Basic block

```
block_type label_one label_two {  
  
  key = value  
  
  embedded_block {  
  
    key = value  
  
  }  
  
}
```

Terraform Syntax - Blocks

#Example block

```
resource "aws_route_table" "route-table" {  
  
  vpc_id = "id928310928"  
  
  route{  
  
    cidr_block = "0.0.0.0/0"  
  
    gateway_id = "id128073987"  
  
  }  
}
```


Terraform Syntax – Object Types

#Different value types

string = "Dosa"

number = 5

bool = true

list = ["Idli", "Dosa"]

map = {name = "Phani", age = 30}

Terraform Syntax - References

#Keyword reference

`var.taco_day`

`aws_instance.taco_truck.name`

`local.taco_toppings.cheeses`

`module.taco_hut.locations`

#Interpolation

`taco_name = "neds-${var.taco_type}"`

Terraform Syntax - References

#Strings, numbers, and bools

local.taco_count #returns the number

#Lists and maps

local.taco_toppings[2] #returns element 3

local.taco_map["likes-tacos"] #returns value at keyname

#Resource values

var.region #returns us-east-1

data.aws_availability_zones.azs.names[1] #returns 2nd
AZ

Scenario

- So far we have instances in multiple availability zones connected through load balancer with a public dns name
- Now the developer wants to add s3 bucket to the configuration.
The S3 bucket is to store the website files and the two instances pull the information from S3 and publish it
- The other thing that the developer would like is to do is to take the logs from the web servers and store them in the s3 buckets for long retention.
- Provide read permissions to instances and write permissions to S3 bucket

Terraform Provisioners



Last resort

Local or remote

Creation or destruction

Multiple provisioners

What if it all goes wrong?

Provisioner Example

```
provisioner "file" {  
  connection {  
    type = "ssh"  
    user = "root"  
    private_key = var.private_key  
    host = var.hostname  
  }  
  source = "/local/path/to/file.txt"  
  destination = "/path/to/file.txt"  
}
```

Provisioner Example

```
provisioner "local-exec" {  
    command = "local command here"  
}  
  
provisioner "remote-exec" {  
    scripts = ["list", "of", "local", "scripts"]  
}
```


Summary

- Terraform provisioners
- Syntax and object types
- S3 buckets, tags, and more!