HashiCorp

Terraform

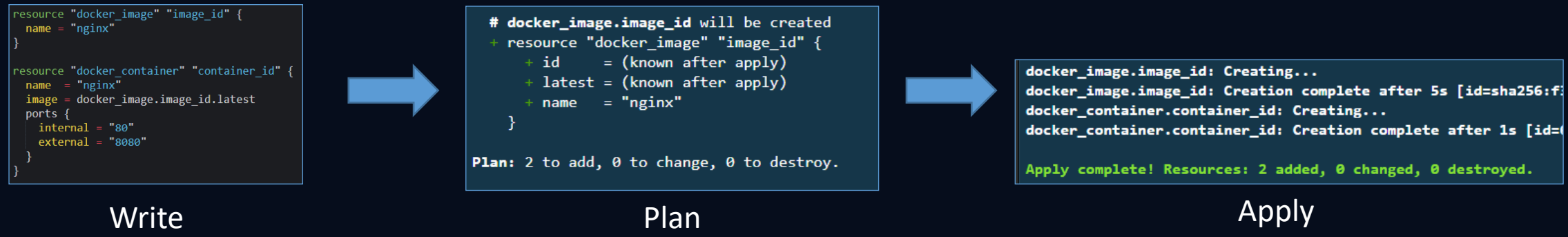
How Terraform Works:

- Written in Golang
- Interfaces with the API of the “provider”
- Create
- Read
- Update
- Delete

```
resource "docker_image" "nodered_image" {
  name = "nodered/node-red:latest"
}
```

= *docker pull nodered/node-red:latest*

Core Terraform Workflow:



Write

Plan

Apply

Terraform State

- Stores information about the current environment
- Is created based on the configuration files and any changes are committed to the infrastructure via the API
- Only knows about resources created by it. If those resources are missing, it can replace, but cannot see other resources.


```
{
  "version": 4,
  "terraform_version": "0.14.7",
  "serial": 1,
  "lineage": "a23576b6-f870-e0e5-31d1-b41736e86628",
  "outputs": {},
  "resources": [
    {
      "mode": "managed",
      "type": "docker_image",
      "name": "nodered_image",
      "provider": "provider[\"registry.terraform.io/terraform-providers/docker\"]",
      "instances": [
        {
          "schema_version": 0,
          "attributes": {
            "id": "sha256:c060f9cb7fd5a4375549f954c0bfac42107094f879a33ab27118749206c42bb0nodered/node-red:latest",
            "keep_locally": null,
            "latest": "sha256:c060f9cb7fd5a4375549f954c0bfac42107094f879a33ab27118749206c42bb0",
            "name": "nodered/node-red:latest",
            "pull_trigger": null,
            "pull_triggers": null
          },
          "sensitive_attributes": [],
          "private": "bnVsbA=="
        }
      ]
    }
  ]
}
```

IaC Workflow:

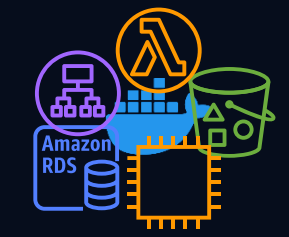
```
resource "docker_image" "image_id" {
  name = "nginx"
}

resource "docker_container" "container_id" {
  name  = "nginx"
  image = docker_image.image_id.latest
  ports {
    internal = "80"
    external = "8080"
  }
}
```

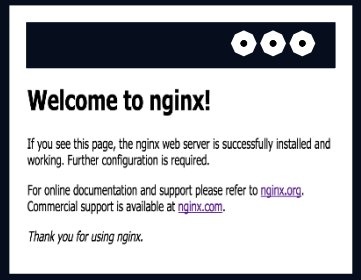
Terraform Code

Git Repository

CICD Tools



Infrastructure



Application

Idempotence

- Can run code as many times as you like while still maintaining the resources specified.
- One reason the “local-exec” provisioner isn’t recommended.
- Isn’t always true. You should ALWAYS verify your plan before applying infrastructure in production.

HashiCorp
Terraform