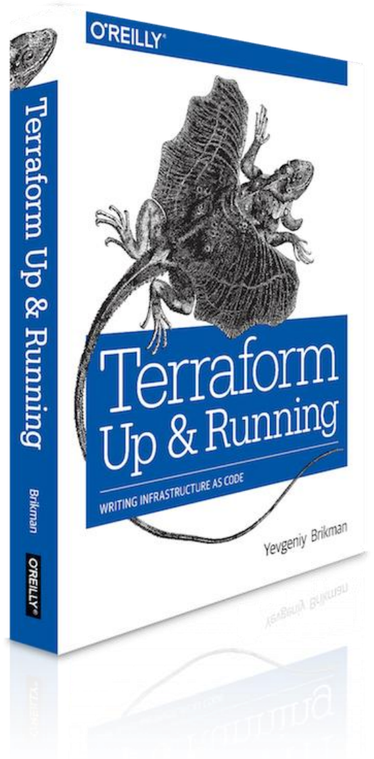
Overview

- What's Terraform
  - HashiCorp (DevOps Unicorn)
  - What's Infrastructure as Code
  - Why Terraform
  - How to use it
  - How Terraform works
  - Terraform commands
  - How Terraform looks like
- Demo: (AWS+Terraform+Ansible)
  - Provisioning a EC2 Instance
  - Provisioning a EC2 with Docker
- Terraform Certification



aws

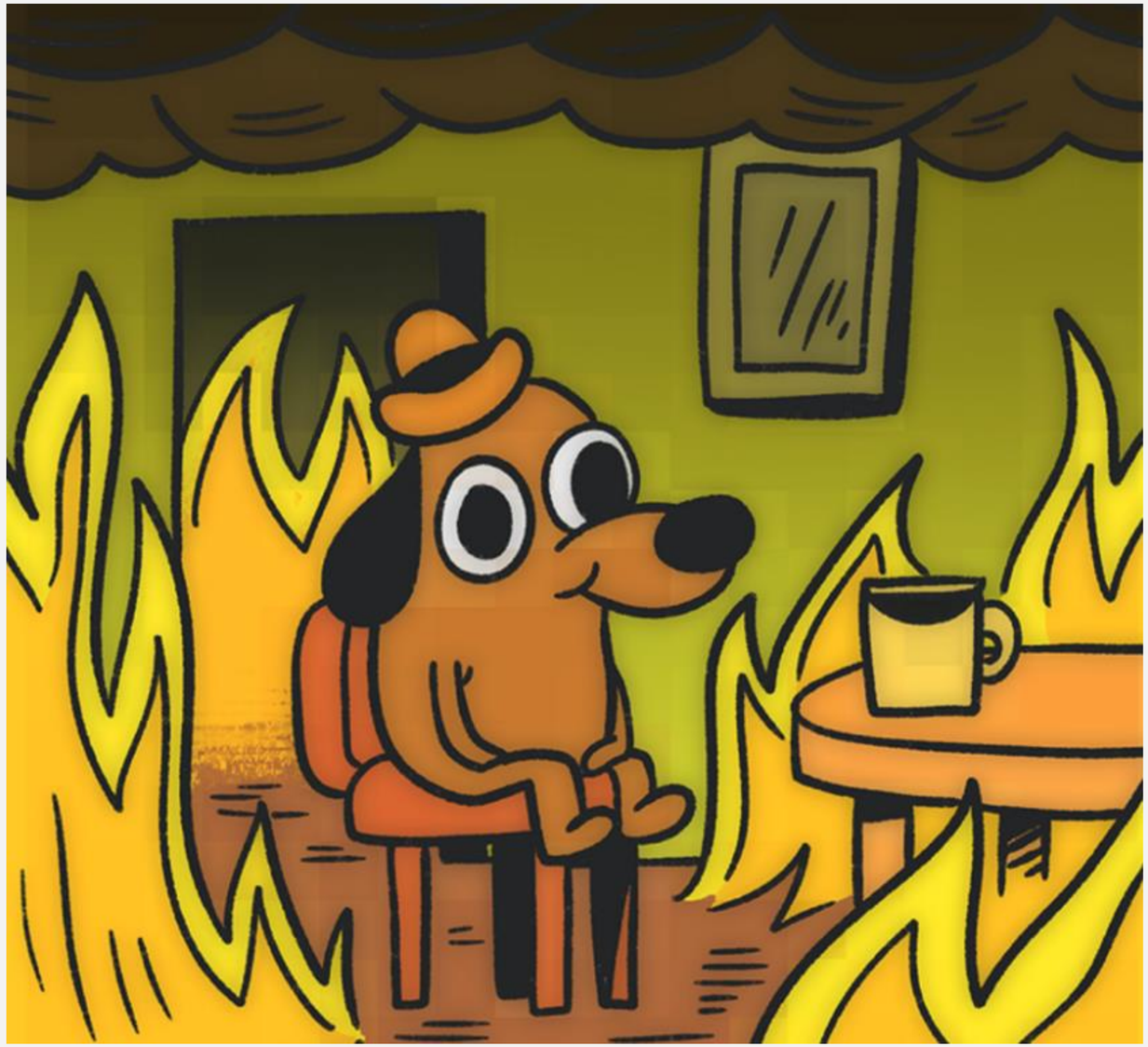
Terraform: Cloud Infrastructure Automation

—

Amaury Borges Souza
Linux System Administrator

Provisioning in the past...

- Manual process (Shell Script)
- Extra knowledge required...
- Many human errors
- DevOps culture didn’t was ready
- There was deploy on Friday
- Infrastructure is not versioned
- Infrastructure as Code not exist
- Communication (Dev and Ops)

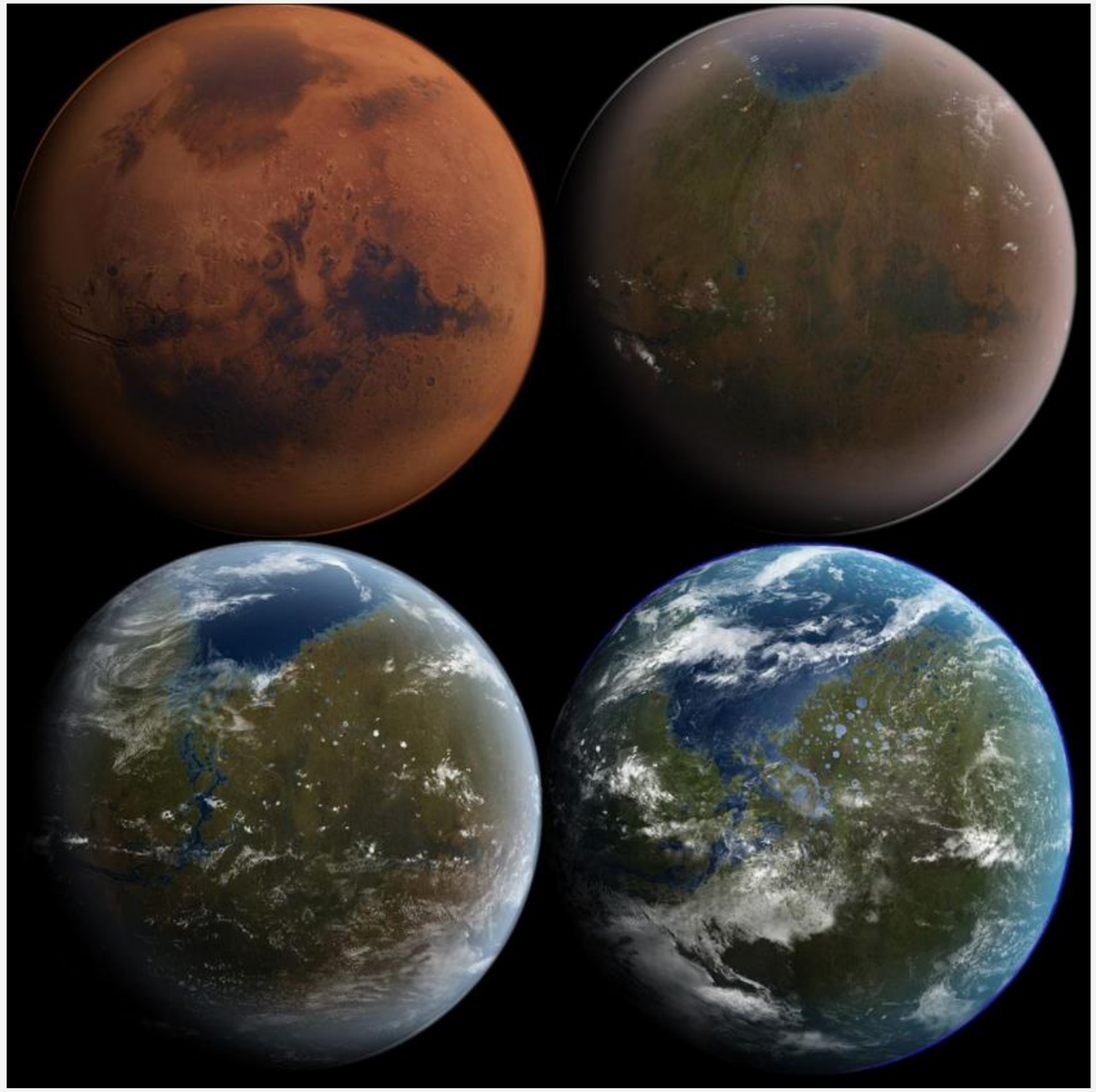


Provisioning today

- DevOps culture is strong
- IaC (Infrastructure as Code)
- Dev and Ops working together
- Terraform, Ansible, AWS, GCP, IBM
- DevOps tools helps
- Infrastructure versioned (Git)
- Better communication
- More automations from all sides

“Terraforming (literally, “Earth Shaping”), is the hypothetical process of deliberately modifying the atmosphere, temperature, surface topography or ecology of a planet, moon, or other body to be similar to the environment of Earth to make it habitable by Earth like-life. **Terraform is a tool for building, changing and versioning the infrastructure.**”

Wikipedia



“Terraform is an open-source infrastructure as code software tool that provides a consistent CLI workflow to manage hundreds of cloud services. **Terraform codifies cloud API’s into declarative configuration files.**”

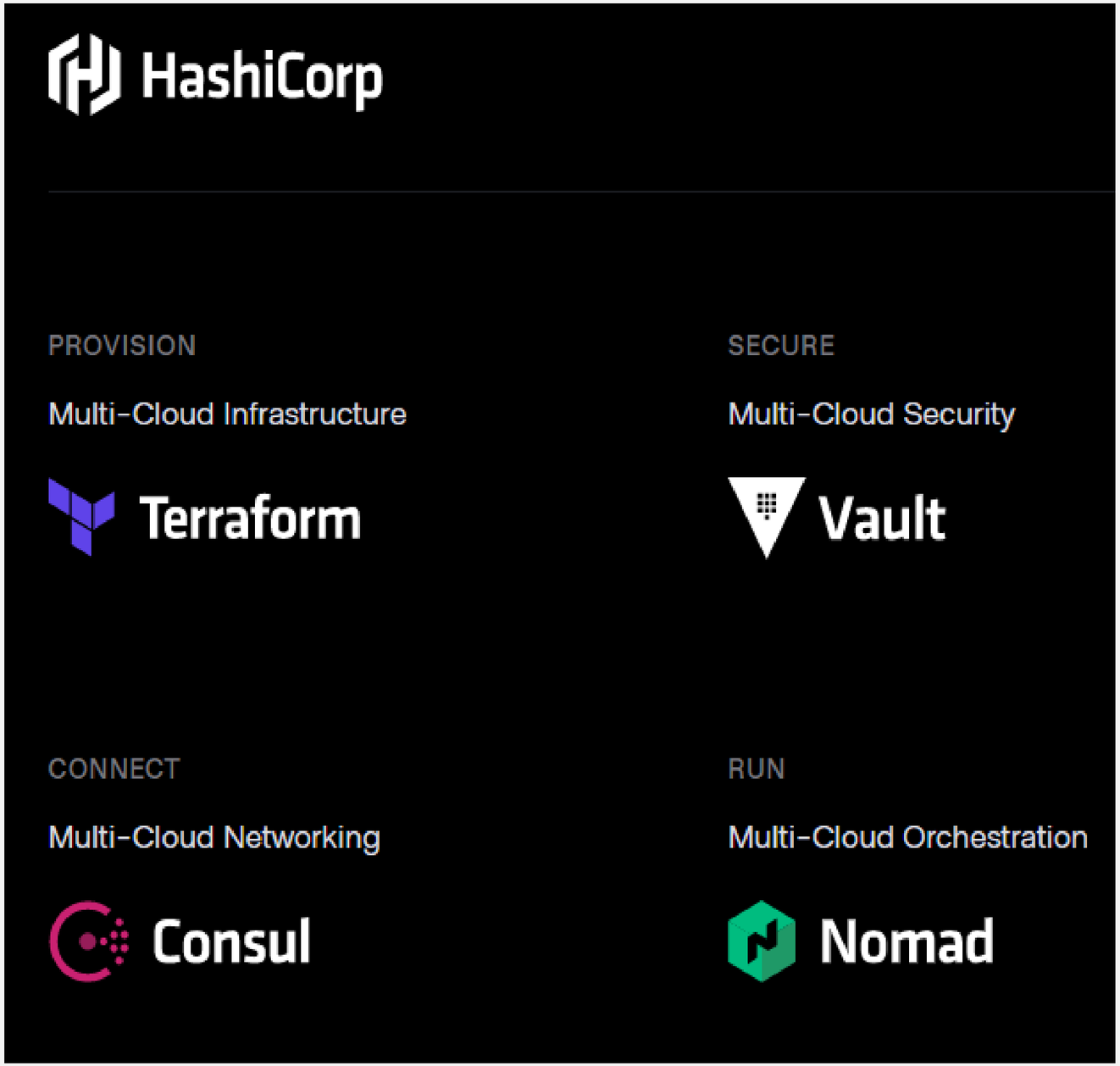
HashiCorp

HashiCorp

one of the most famous companies that supply products used by the DevOps community, where it has as tools, **Terraform, Vault, Consul, Nomad, Vagrant, Packer,** etc.

HashiCorp Terraform Cloud named a **`finalist for 2020 CRN Tech Innovator Award`**.

HashiConf Global: an online HashiCorp Community conference. Oct 19-20, 2021



Features…

- Popular infrastructure providers (IBM,AWS, GCP, Azure)
- Execution Plans
- Resource Graph
- **It’s not a configuration management tool**
- Versioning the infrastructure
- Open Source and declarative
- Idempotency
- HCL Language (.tf)

IBM **Cloud**

Google Cloud Platform

aws

kubernetes

```
resource "aws_instance" "iac_in_action" {
  ami               = var.ami_id
  instance_type     = var.instance_type
  availability_zone = var.availability_zone

  // dynamically retrieve SSH Key Name
  key_name = aws_key_pair.iac_in_action.key_name

  // dynamically set Security Group ID (firewall)
  vpc_security_group_ids = [aws_security_group.iac_in_action.id]

  tags = {
    Name = "Terraform-managed EC2 Instance for IaC in Action"
  }
}
```

What’s Infrastructure as Code?

Basically, infrastructure as code (IaC) is the managing and provisioning of infrastructure through code instead of through manual process.

Red Hat

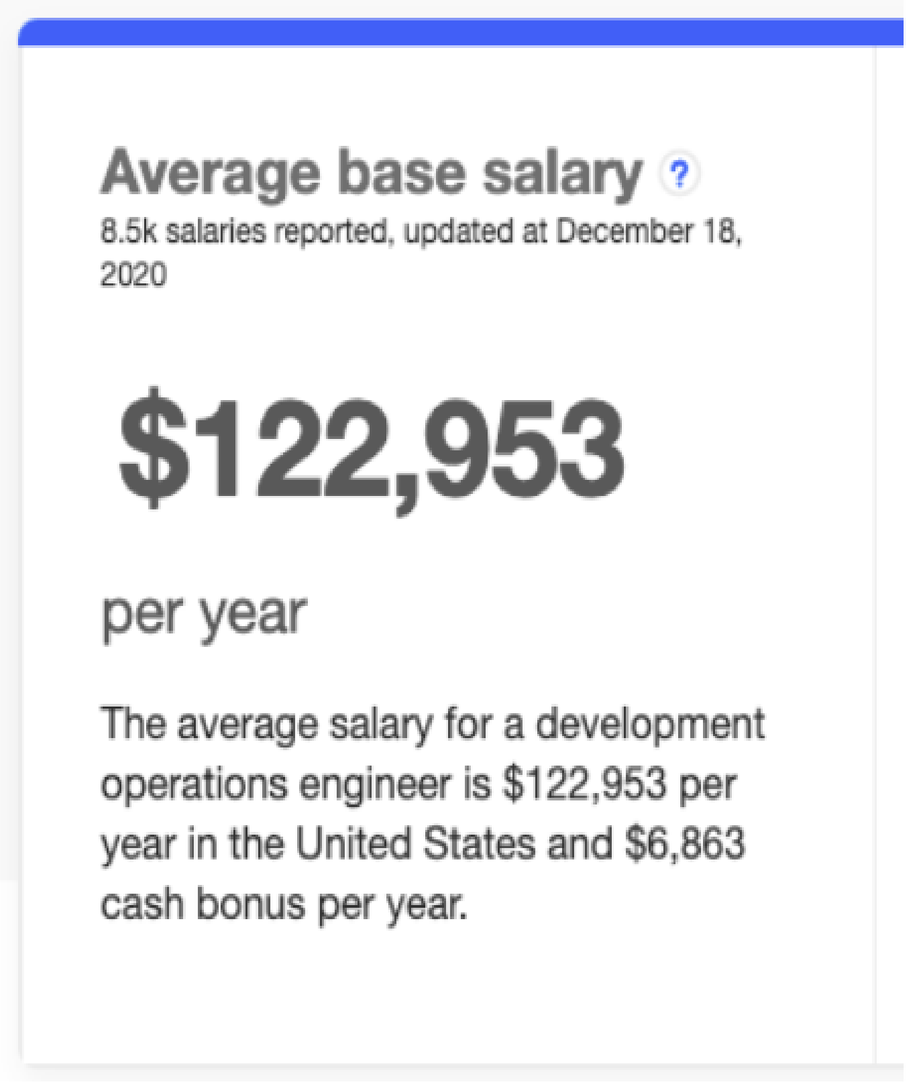
Why Terraform

- The average base salary of a DevOps/Automation Engineer is $167,292 in San Francisco and $122,953 in United States (2020 numbers)

- Terraform has gained a lot in popularity over the last years.

- It is now one of the most sought-after skills in this field.

- It’s a tool that you will need to master when provisioning and cloud environment.



More about Terraform…

"Codification allows infrastructure changes to be automated, while keeping the definition human readable. Automated tooling allows operators to increase their productivity, move quicker and reduce human error."

Multi-Cloud

Popular Infrastructure Providers
(IBM Cloud, AWS, Azure, GCP)

Configuration Files (.tf)

Execution Plans

- **450,000+ Commits**
- **4,000+ Modules**
- **500+ Providers**

Open source projects benefit from the scrutiny of a broad and diverse user base. Keeping the code available helps empower the community of users while also providing an easy mechanism for feedback, improvement, and customization.

How to use it

- Download the appropriate package:

https://www.terraform.io/downloads.html

- Now, unzip the file

- Add terraform executable to your $PATH

- Run it:

$ terraform --version

```
Terraform v0.12.29
+ provider.aws v3.25.0

Your version of Terraform is out of date! The latest version
is 0.14.9. You can update by downloading from https://www.terraform.io/downloads.html
```

macOS
64-bit

FreeBSD
32-bit | 64-bit | Arm

Linux
32-bit | 64-bit | Arm | Arm64

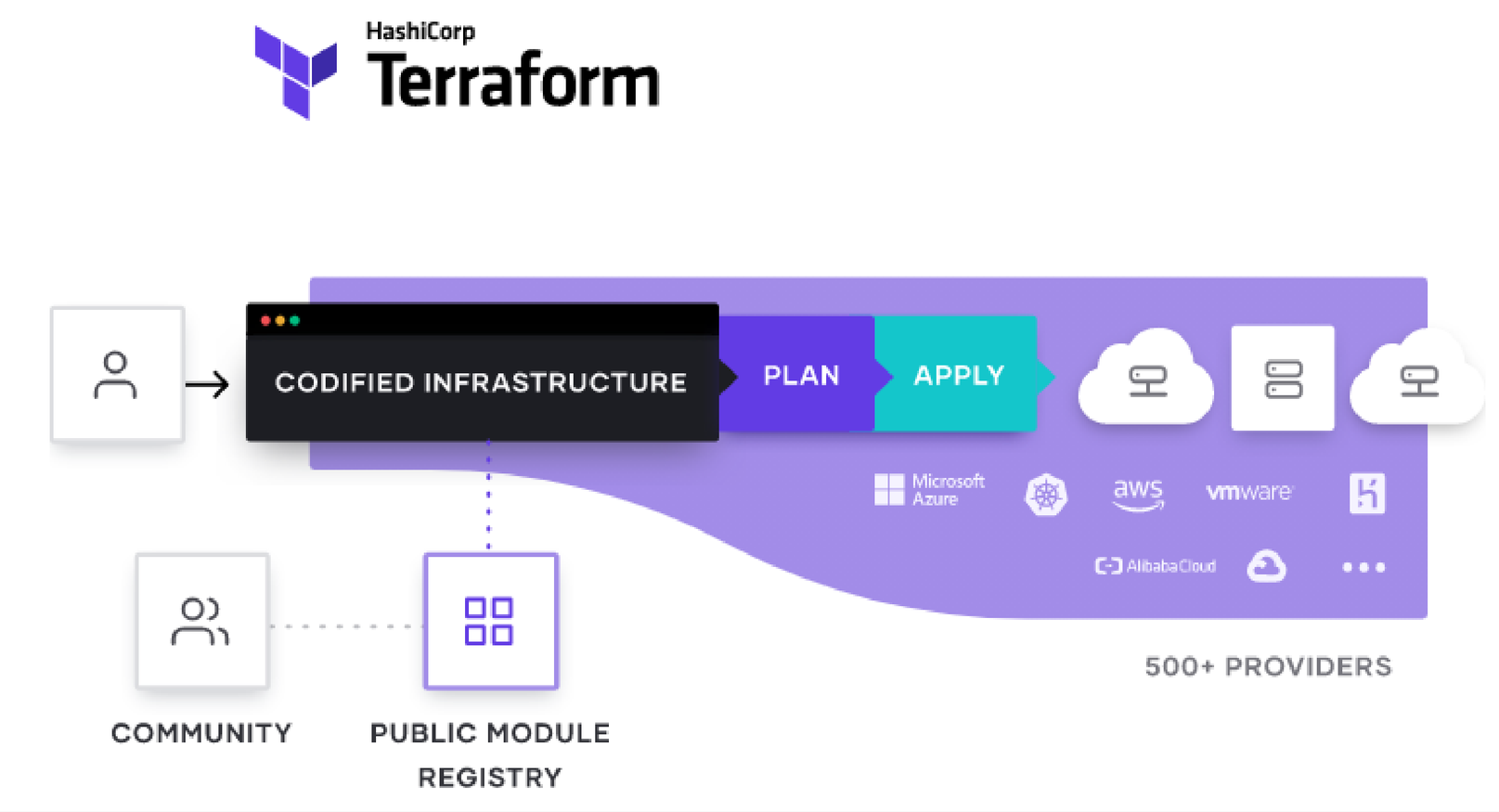
OpenBSD
32-bit | 64-bit

Solaris
64-bit

Windows
32-bit | 64-bit

How Terraform works

- Allows infrastructure to be expressed as code.
- Collaborate on infrastructure as code
- Self-Service infrastructure
- Uses HCL (HashiCorp Configuration Language)
- Provides an execution plan
- Policy and Governance
- Apply resources quicky
- IaaS, PaaS, SaaS



Terraform: main commands

$ terraform init

Initialize a Terraform working directory.

$ terraform plan

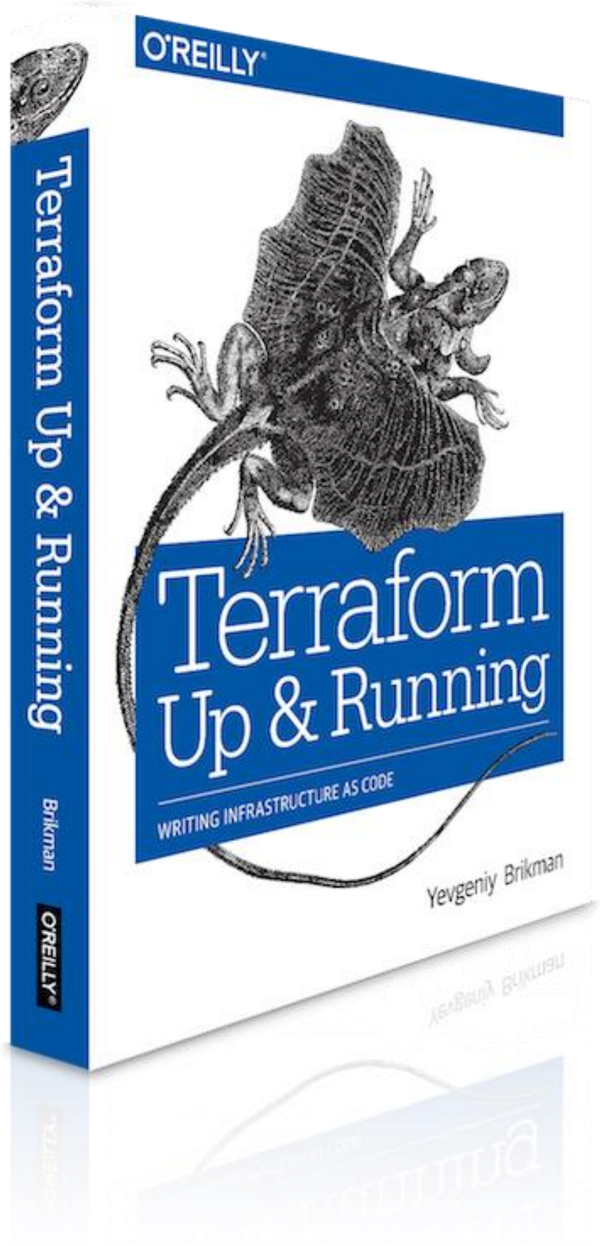
Generate and show an execution plan.

$ terraform apply

Builds or changes infrastructure.

$ terraform destroy

Destroy Terraform-managed infrastructure.



Terraform: other features

1 Providers

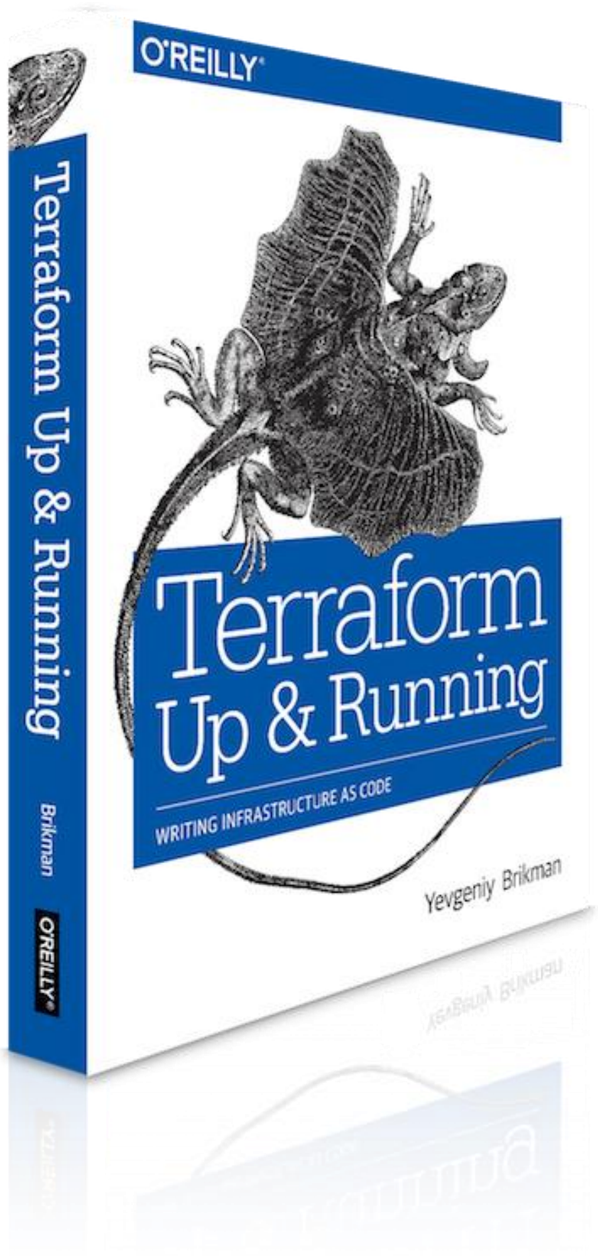
Interact with cloud providers, SaaS providers and other API's.

2 Resources

Each resource block describes one or more infrastructure objects, such as, compute instances, virtual networks, etc.

3 Modules

Consist of a collection of ".tf" file kept together in a directory.

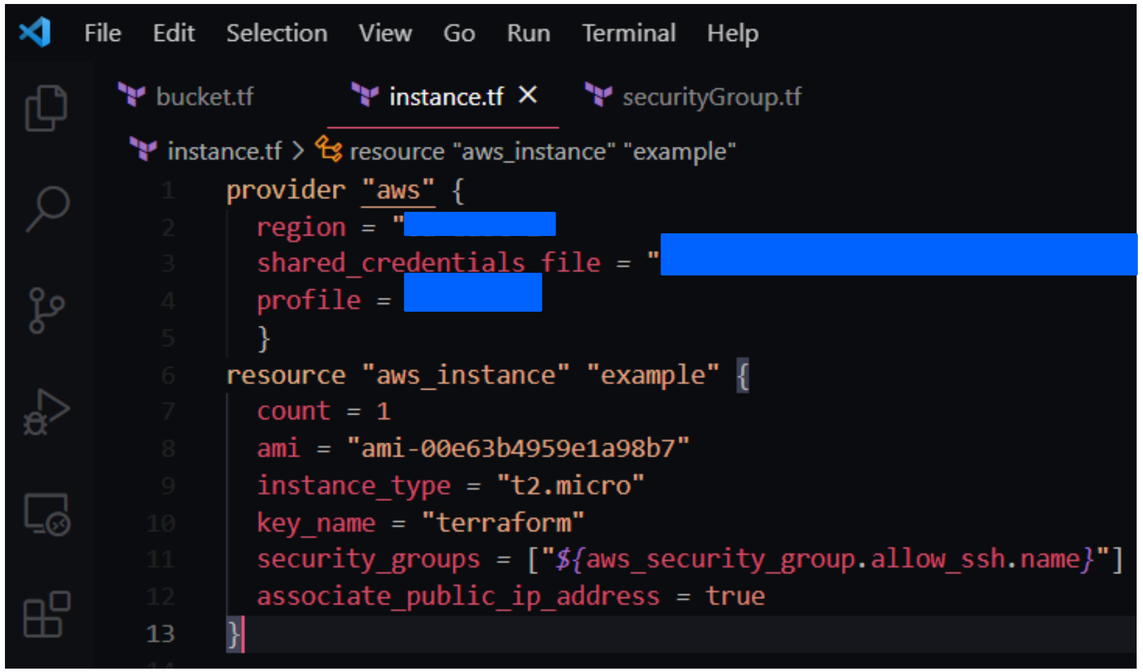


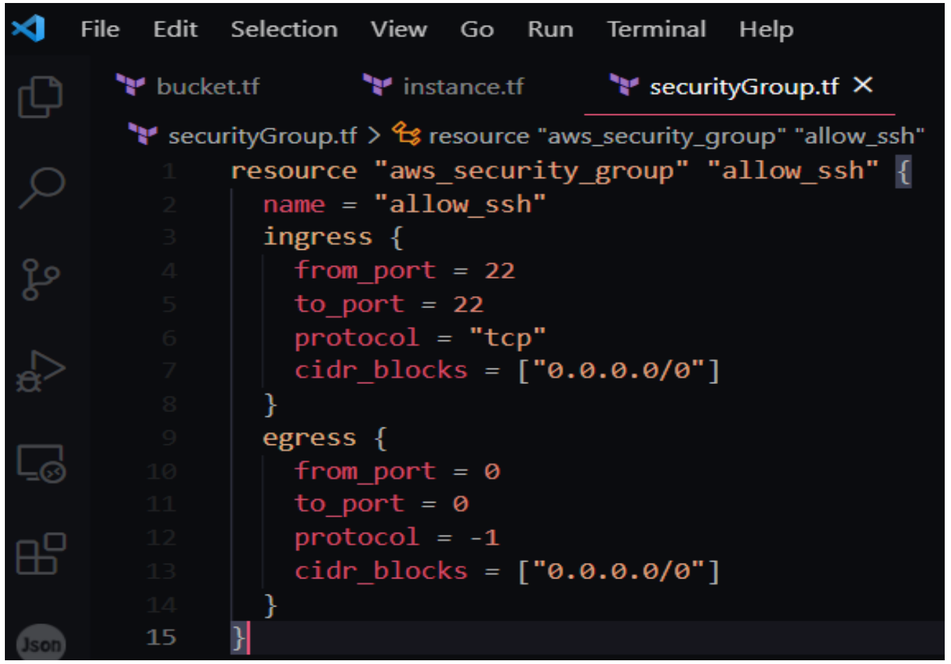
How Terraform looks like

Uses the HCL (HachiCorp Configuration Language) to write the configuration files ".tf" to provision your infrastructure services.

https://www.terraform.io/docs/language/syntax/configuration.html

It is not JSON, but it's accepts JSON format.





Demo

‘Talk is cheap. Show me the code.’
Linus Torvalds

Provisioning an EC2 instance on AWS 🚀

HashiCorp Terraform

aws

Adding a resource on infrastructure (EC2 Instance)

EXPLORADOR

TERRAFORM
- aws\EC2
  - 01
    - .terraform
    - .terraform.lock.hcl
    - provider.tf
    - terraform.tfstate
    - terraform.tfstate.backup
    - variables.tf

Bem-vindo(a) | provider.tf | variables.tf

aws > EC2 > 01 > provider.tf > resource "aws_instance" "server2" > instance_type

```tf
provider "aws" {
  region = var.region
}

resource "aws_instance" "server2" {
  ami           = "ami-09e67e426f25ce0d7"
  instance_type = "t2.micro"
}
```

EXPLORADOR

TERRAFORM
- aws\EC2
  - 01
    - .terraform
    - .terraform.lock.hcl
    - provider.tf
    - terraform.tfstate
    - terraform.tfstate.backup
    - variables.tf

Bem-vindo(a) | provider.tf | variables.tf

aws > EC2 > 01 > variables.tf > variable "region"

1 reference

```tf
variable "region" {
  description = "define what region"
  default     = "us-east-1"
}
```

Changing the infrastructure (Tags on EC2 Instance)

EXPLORADOR

TERRAFORM
- aws\EC2
  - 01
  - 02
    - .terraform
    - .terraform.lock.hcl
    - provider.tf
    - terraform.tfstate
    - terraform.tfstate.backup
    - variables.tf

provider.tf ...\01 | provider.tf ...\02 | variables.tf

aws > EC2 > 02 > provider.tf > resource "aws_instance" "server2" > tags

```tf
provider "aws" {
  region = var.region
}

resource "aws_instance" "server2" {
  ami           = "ami-09e67e426f25ce0d7"
  instance_type = "t2.micro"

  tags = {
    Name = "test machine"
  }
}
```

EXPLORADOR

TERRAFORM
- aws\EC2
  - 01
  - 02
    - .terraform
    - .terraform.lock.hcl
    - provider.tf
    - terraform.tfstate
    - terraform.tfstate.backup
    - variables.tf

provider.tf | variables.tf ...\02 | variables.tf ...\01

aws > EC2 > 02 > variables.tf > variable "region"

1 reference

```tf
variable "region" {
  description = "define what region"
  default     = "us-east-1"
}
```

Changing the infrastructure (Tags on EC2 Instance)

aws Services ▼ Search for services, features, marketplace products, and docs [Alt+S] Support ▼

New EC2 Experience
Tell us what you think

- Painel EC2
- Eventos
- Tags
- Limites
- **Instâncias**
  - Instâncias New
  - Tipos de instância
  - Modelos de execução
  - Solicitações spot
  - Savings Plans
  - Instâncias reservadas New
  - Hosts dedicados
  - Instâncias programadas
  - Reservas de capacidade
- **Imagens**
  - AMIs
- **Elastic Block Store**
  - Volumes
  - Snapshots

Instâncias (1/4) Informações

Conectar | Estado da instância ▼ | Ações ▼ | Executar instâncias

Filtrar instâncias

1

	Name	ID de instância	Estado da inst...	Tipo de inst...	Verificação de s...	Status do al...	Zona de dispon...	DNS IPv4 públic
☐	–	[redacted]	Interrompido	t2.micro	–	Sem alar...	us-east-1e	–
☐	–	i-0336f9babcc85557d	Encerrado	t2.micro	–	Sem alar...	us-east-1b	–
☐	test machine	i-03b54e2f3f26665ab	Encerrado	t2.micro	–	Sem alar...	us-east-1b	–
☑	test machine	i-0e3a34642afb87439	Executando	t2.micro	Inicializando	Sem alar...	us-east-1b	ec2-34-227-20-2

Instância: i-0e3a34642afb87439 (test machine)

Detalhes | Segurança | Redes | Armazenamento | Verificações de status | Monitoramento | Tags

Resumo da instância Informações

ID de instância
i-0e3a34642afb87439 (test machine)

Endereço IPv6
–

Endereço IPv4 público
34.227.20.238 | endereço aberto

Estado da instância
Executando

Endereços IPv4 privados
172.31.88.194

DNS IPv4 público
ec2-34-227-20-238.compute-1.amazonaws.com | endereço aberto

Terraform + Ansible: Provisioning an EC2 instance on AWS and deploy Docker 🚀

HashiCorp Terraform

ANSIBLE

aws

HashiCorp Certified: Terraform Associate

The Terraform Associate certification is for Cloud Engineers specializing in operations, IT, or development who know the basic concepts and skills associated with open source HashiCorp Terraform.

Candidates will be best prepared for this exam if they have professional experience using Terraform in production but performing the exam objectives in a personal demo environment may also be sufficient.

Prerequisites
- Basic terminal skills.
- Basic understanding of on premises and cloud architecture.

Exam Details

Assessment Type	Multiple choice
Format	Online proctored
Duration	1 hour
Price	$70.50 USD plus locally applicable taxes and fees
Language	English
Expiration	2 years

How about Truist? Does any case to use Terraform?

- Is it possible to manage X snapshots with the Terraform or Ansible?
- How about X? Terraform can substitute the Linux builds on the X environment?
- We already know that **X** uses **AWS**, **be prepared** when we need to **launch resources** on the **Cloud.** 🚀

Thank you

Amaury Borges Souza
Linux System Administrator
—