Infrastructure deployment on AWS using Terraform

Agenda

- Introduction
- What is Terraform
- Infrastructure provisioning challenges
- Why Terraform
- Demo
- Next Steps
- Q & A

What is Terraform

- Tool for provisioning
- Safely and predictably building, changing and versioning infrastructure

Infrastructure provisioning challenges

- Long infrastructure design
- Difficulties in becoming expert in every cloud provider
- Difficult incidents management
- Prisoner of devices vendor / cloud provider
- Limitation of cloud provider availability
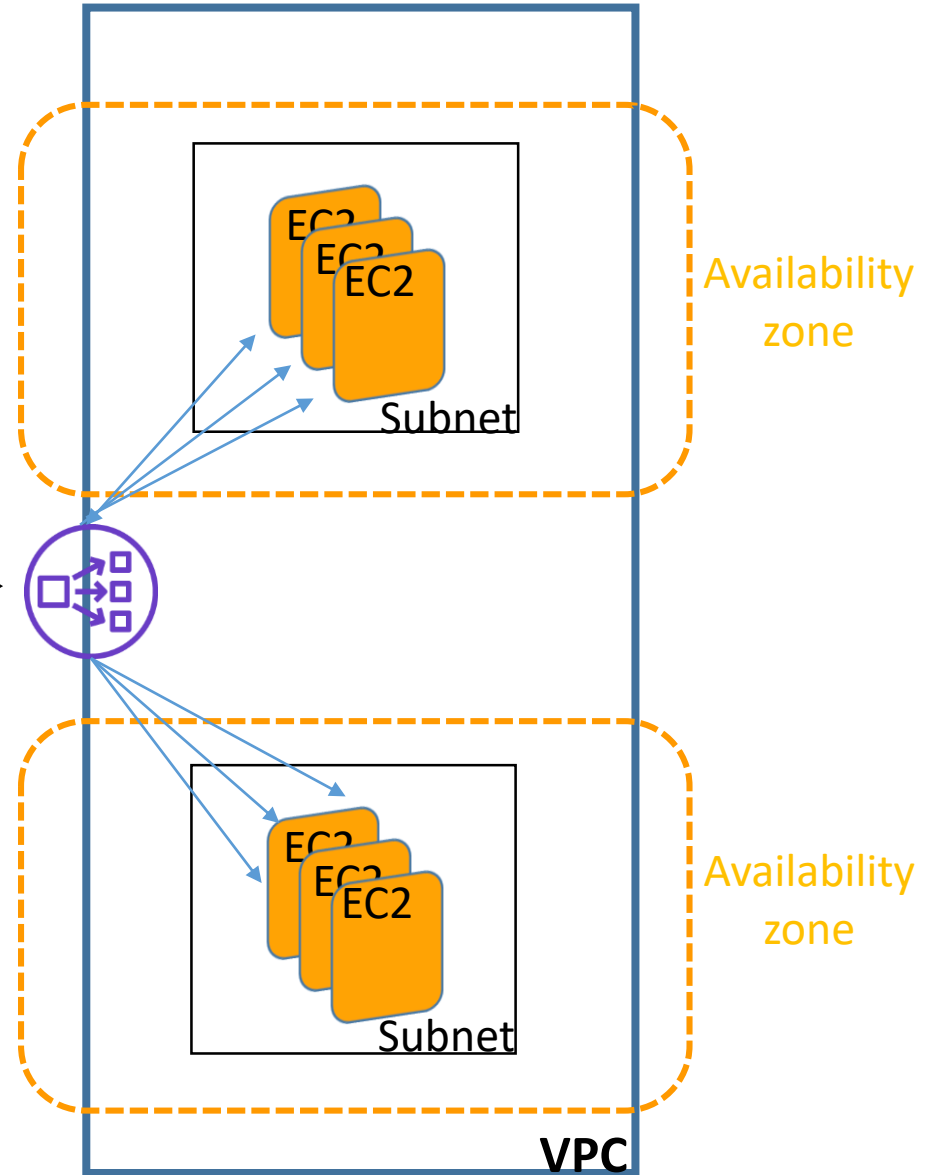
Why Terraform

- Fast Infrastructure provision
- Easy readability of Infrastructure as Code (IaC)
- Certainty of the infrastructure to be provisionned
- Support of many providers including AWS, GCP and Azure
- Multi cloud management and orchestration

Demo

Distributed and highly available infrastructure deployment on AWS using Terraform

```
variable "server_port" {


provider "aws" {
    region = "eu-west-3"
}

resource "aws_instance" "demo" {
  ami="ami-0bb607148d8cf36fb"
  instance_type = "t2.micro"
.
.
.
}
```

Terraform files

Deployed infrastructure on AWS

EC2
EC2
EC2
Subnet

Availability zone

EC2
EC2
EC2
Subnet

Availability zone

VPC

Environment setup

- Download Terraform from *terraform.io/downloads.html*

- Unzip the downloaded terraform file and place "terraform" binary on the PATH

```
$ unzip terraform_"<VERSION>"_linux_amd64.zip
$ mv terraform /usr/bin/terraform
```

- Clone *github.com/ndekwe/terraform* project and access project directory

```
$ git clone https://github.com/ndekwe/terraform.git
$ cd terraform
```
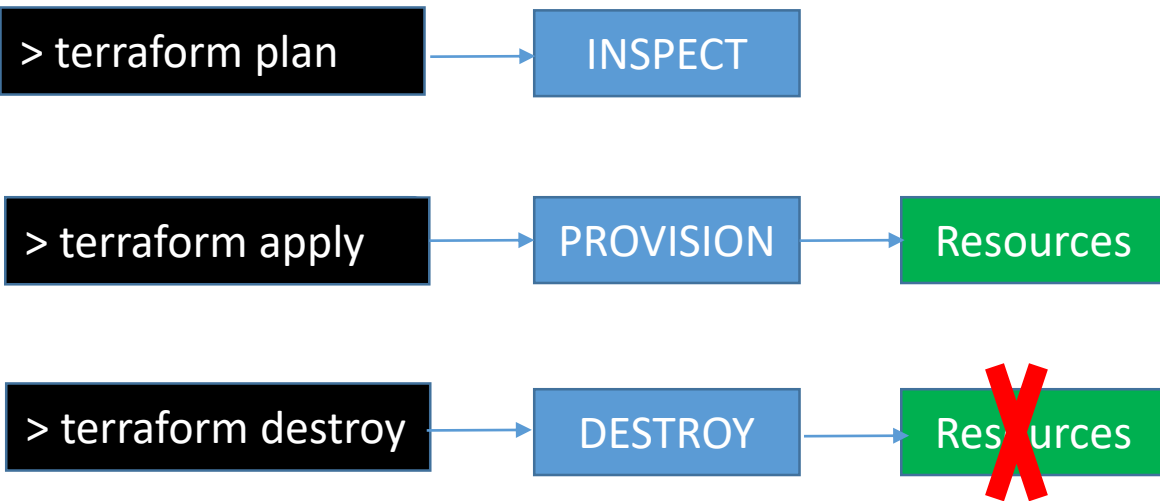
- Setup "AWS access" and "AWS secret access" keys on CLI and download AWS plugin

```
$ export AWS_ACCESS_KEY="Place AWS access key here"
$ export AWS_SECRET_ACCESS_KEY="Place AWS secret access key here"
$ terraform init
```

Demo

More on Terraform

> terraform plan → INSPECT

> terraform apply → PROVISION → Resources

> terraform destroy → DESTROY → Resources ✗

- state
- modules
- graph
- refresh
- provider

Terraform Cloud: https://app.terraform.io/session

Terraform Registry: https://registry.terraform.io/

Next steps

- Learn more:
  - Hashicorp learn portal: **learn.hashicorp.com**
  - Terraform documentation: **terraform.io/docs/index.html**
  - Blog: **hashicorp.com/blog**

- Contribute to the vibrant community: **hashicorp.com/community**

- Join HashiCorp Events: **hashicorp.com/events**
  - HashiConf
  - HashiCorp User Groups (HUGs)
  - Free Workshops
  - Webinars
  - Training

Q & A

Alexis Munyandekwe

@ndekwe

linkedin.com/in/ndekwe

ndekwe   (github.com/ndekwe/terraform)