Module 1

Introducing Infrastructure as Code and Terraform

Agenda

- Infrastructure as Code
- Introducing Terraform
- Comparing Vendor tools to Terraform
- Terraform Basics
- Lab 1

HashiCorp
Terraform

Infrastructure as Code

- Traditional Datacentre
- Software defined X
- Removing the human
- Disposability

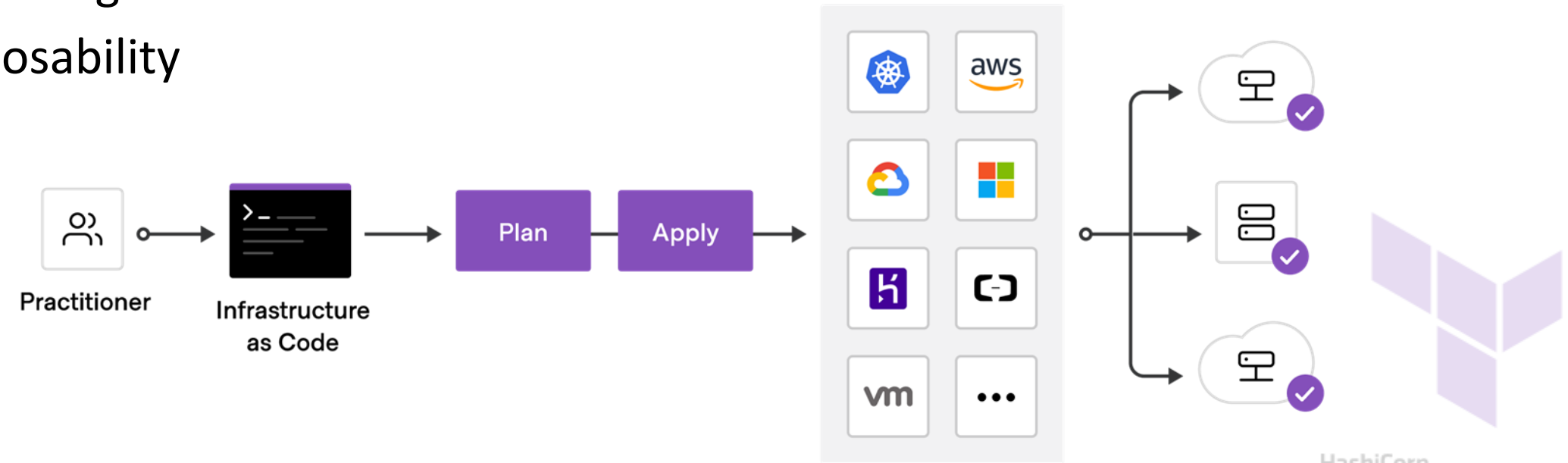


Image courtesy of learn.hashicorp.com

HashiCorp Terraform

Introducing Terraform

- Infrastructure as Code tool
- Can manage on-prem and cloud
- Is cloud agnostic
- Supports both JSON and HCL

HashiCorp
Terraform

Comparing Vendor tools to Terraform

- AWS
  CloudFormation

- Azure
  ARM Templates

- GCP
  Deployment Manager

HashiCorp
Terraform

Terraform Basics

The Terraform Workflow:

Scope - What are you trying to achieve?

Write - Author infrastructure as code.

Init - Initialize the code stack.

Plan - Preview changes before applying.

Apply - Provision reproducible infrastructure.

```
terraform {
  required_providers {
    docker = {
      source = "kreuzwerker/docker"
      version = "~> 2.21.0"
    }
  }
}

provider "docker" {}

resource "docker_image" "Apache_web" {
  name         = "httpd:latest"
  keep_locally = false
}

resource "docker_container" "web" {
  image = docker_image.Apache_web.image_id
  name  = "Web-demo"
  ports {
    internal = 80
    external = 8080
  }
}
```

Terraform Basics

The Core Terraform Workflow:

Write - Author infrastructure as code.

Plan - Preview changes before applying.

Apply - Provision reproducible infrastructure.

```
terraform {
  required_providers {
    docker = {
      source = "kreuzwerker/docker"
      version = "~> 2.21.0"
    }
  }
}

provider "docker" {}

resource "docker_image" "Apache_web" {
  name         = "httpd:latest"
  keep_locally = false
}

resource "docker_container" "web" {
  image = docker_image.Apache_web.image_id
  name  = "Web-demo"
  ports {
    internal = 80
    external = 8080
  }
}
```

Terraform Basics

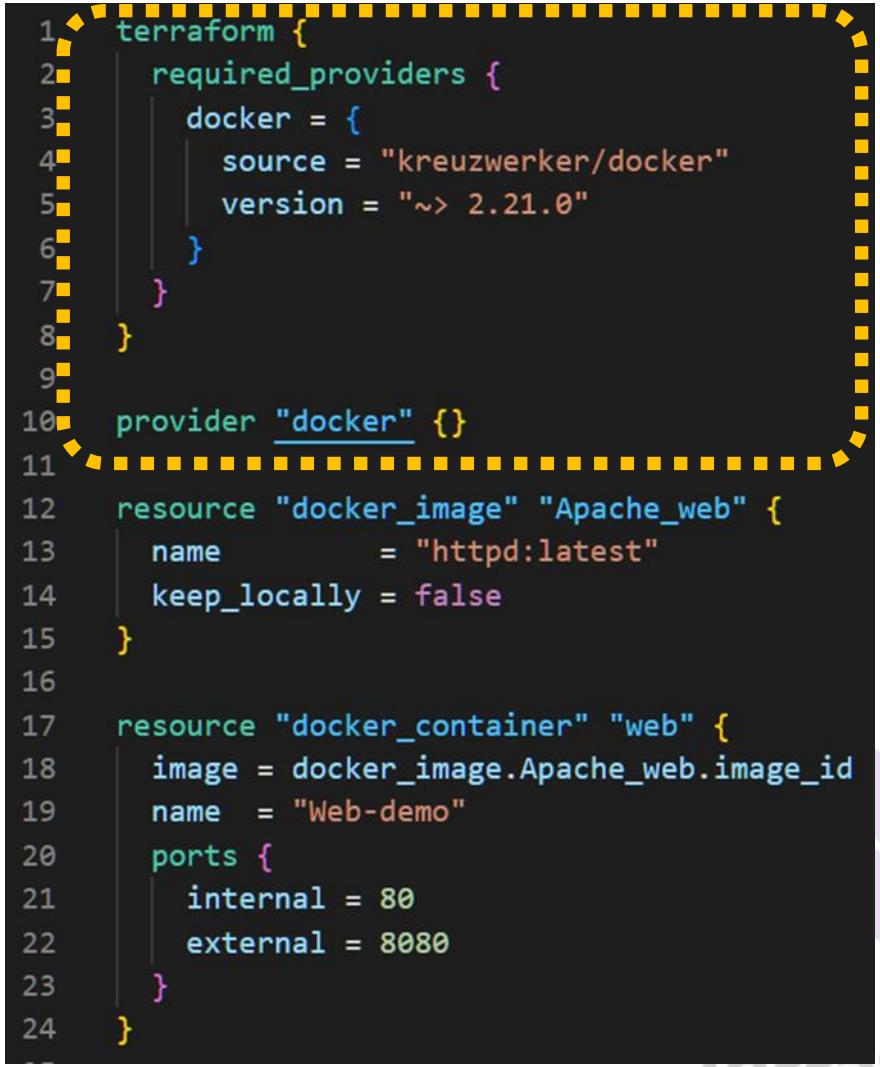
Write

Provider Information

Workflow Blocks

Plan

Apply



```
terraform {
  required_providers {
    docker = {
      source = "kreuzwerker/docker"
      version = "~> 2.21.0"
    }
  }
}

provider "docker" {}

resource "docker_image" "Apache_web" {
  name         = "httpd:latest"
  keep_locally = false
}

resource "docker_container" "web" {
  image = docker_image.Apache_web.image_id
  name  = "Web-demo"
  ports {
    internal = 80
    external = 8080
  }
}
```

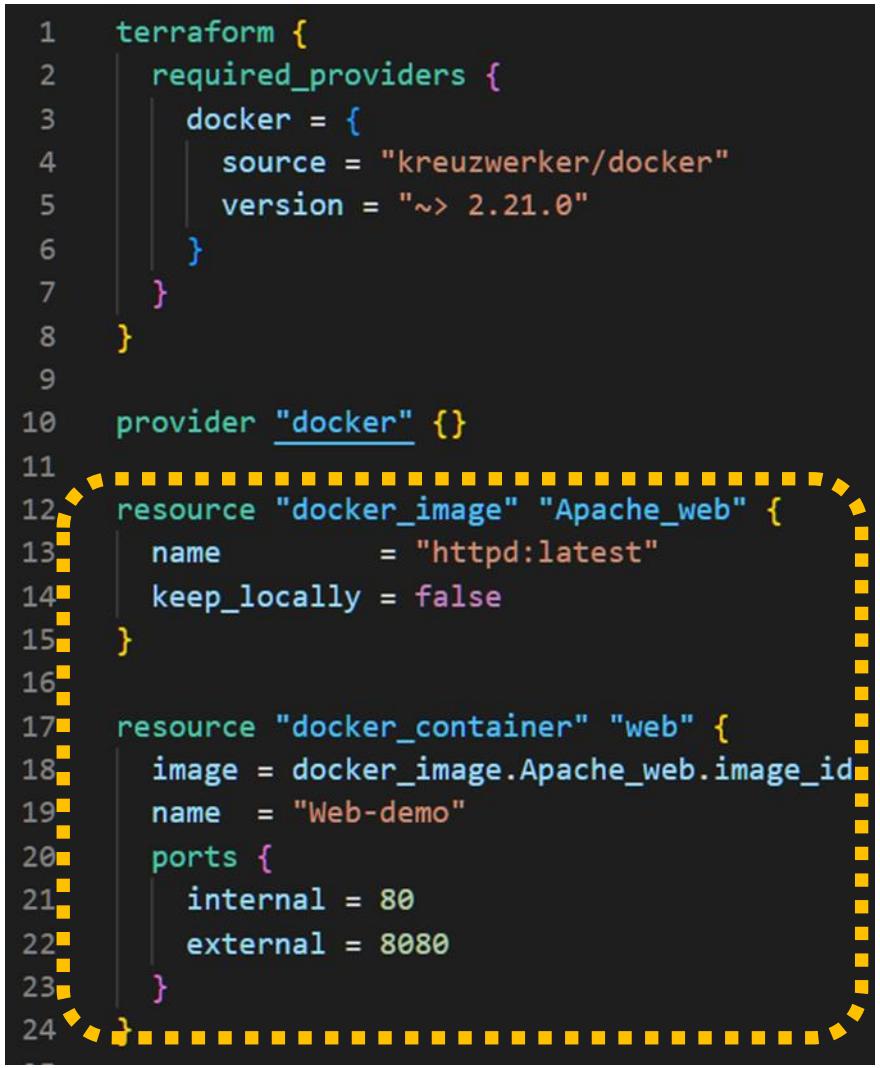
Terraform Basics

Write

- Provider Information
- Workflow Blocks

Plan

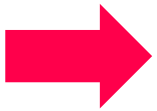
Apply



```
terraform {
  required_providers {
    docker = {
      source = "kreuzwerker/docker"
      version = "~> 2.21.0"
    }
  }
}

provider "docker" {}

resource "docker_image" "Apache_web" {
  name         = "httpd:latest"
  keep_locally = false
}

resource "docker_container" "web" {
  image = docker_image.Apache_web.image_id
  name  = "Web-demo"
  ports {
    internal = 80
    external = 8080
  }
}
```

Terraform FMT

```
terraform {
required_providers {
docker = {source = "kreuzwerker/docker"
version = "~> 2.21.0"}
}
}
provider "docker" {}
resource "docker_image" "Apache_web"{
name = "httpd:latest"
keep_locally = false
}
resource "docker_container" "web"{
image = docker_image.Apache_web.image_id
name  = "Web-demo"
ports {
internal = 80
external = 8080
}
}
```



```
terraform {
  required_providers {
    docker = { source = "kreuzwerker/docker"
    version = "~> 2.21.0" }
  }
}
provider "docker" {}
resource "docker_image" "Apache_web" {
  name         = "httpd:latest"
  keep_locally = false
}
resource "docker_container" "web" {
  image = docker_image.Apache_web.image_id
  name  = "Web-demo"
  ports {
    internal = 80
    external = 8080
  }
}
```


Note: ***we do not TAB, we indent two spaces***

Lab Group Discussion & Explore

Scoping

- What Type of Resources are you going to create ?
- What specific Properties do we need to consider for these resources ?

Workflow

- Are there any Implicit or Explicit dependencies ?
- What TF *Provider*, *Resource,* and *Modules* are required ?

Research

- Identify the TF documentation that will assist in developing the solution

HashiCorp Terraform

Lab 1

Creating a Local Resource

In this lab you will deploy a Docker image and container as local resources on an EC2 instance

HashiCorp Terraform

Any questions…

Terraform